Prediction of colligative effects in the system Water + NaCl through Machine Learning

Jorge Eduardo Londoño Arango

Thesis submitted in partial fulfillment of the requirements for the degree of:
Master in Engineering – Chemical Engineering

Advisor:
Ph.D Javier Ignacio Carrero Mantilla

Research Group:
Computational physical chemistry

Universidad Nacional de Colombia
Facultad de Ingeniería y Arquitectura, Departamento de Ingeniería Química
Manizales, Colombia
2023

Predicción de los efectos coligativos en el sistema Agua + NaCl mediante Machine Learning

Jorge Eduardo Londoño Arango

Tesis presentada como requisito parcial para optar al título de:
Magister en Ingeniería - Ingeniería Química

Director(a):
Profesor Javier Ignacio Carrero Mantilla

Grupo de Investigación:
Fisicoquímica computacional

Universidad Nacional de Colombia
Facultad de Ingeniería y Arquitectura, Departamento de Ingeniería Química
Manizales, Colombia
2023

La vida es ese encuentro y desencuentro constante, esa pérdida y esa acumulación inconsciente y no buscada. Esas personas que vemos o conocemos y que creemos estarán allí siempre y que se irán diluyendo en la distancia y el tiempo, tal como los tíos catalanes. A la vida no hay manera de darle la vuelta, va yendo, tal como transcurre el camino.

Pablo Felipe Arango.

La vida no es la vida que vivimos, la vida es el honor, es el recuerdo. Por eso hay muertos que en el mundo viven, y hombres que viven en el mundo, muertos.

Antonio Muñoz Feijoo.

¡Ay! ¡Cuántas veces al reír se llora! ¡Nadie en lo alegre de la risa fíe, porque en los seres que el dolor devora el alma llora cuando el rostro ríe!.

Juan de Dios Peza.

Statement

I affirm that I have carried out this thesis autonomously and with the sole assistance of permissible resources mentioned within the thesis itself. Any passages, whether verbatim or figurative, taken from published or unpublished texts, have been acknowledged in this work. No part of this work has been used in any other thesis.

Me permito afirmar que he realizado la presente tesis de manera autónoma y con la única ayuda de los medios permitidos y no diferentes a los mencionados en la propia tesis. Todos los pasajes que se han tomado de manera textual o figurativa de textos publicados y no publicados, los he reconocido en el presente trabajo. Ninguna parte del presente trabajo se ha empleado en ningún otro tipo de tesis.

Manizales, Caldas, Julio de 2023

Jorge Eduardo Londoño Arango.

Agradecimientos

A Isabel Sofía Arias Arango, mi mayor fuente de inspiración y el motor que me impulsa a seguir adelante. Gracias por ser mi roca y mi apoyo incondicional en cada etapa de mi vida y especialmente en este proyecto tan importante para mí.

A Samuel y Simón Londoño Valencia, mis hermanos y mis mejores amigos. Gracias por enseñarme que el valor de la amistad y el amor fraternal son invaluables. Su presencia en mi vida ha sido una bendición y me han dado la fortaleza necesaria para superar cualquier obstáculo.

A Maria Victoria Arango Tobón y Orlando Londoño Betancourt, mis padres y mis confidentes. Gracias por brindarme su amor incondicional y su apoyo inquebrantable. Sin su ayuda, este proyecto no habría sido posible. Son los mejores amigos que he tenido y los amo con todo mi corazón.

A Javier Ignacio Carrero Mantilla, mi tutor y guía en este proyecto. Gracias por su sabiduría, paciencia y dedicación. Ha sido un honor trabajar con alguien tan apasionado y comprometido con su trabajo. Sus consejos y enseñanzas serán invaluables en mi carrera profesional.

A los ingenieros Santiago Ramírez Ramírez y Sergio Arango Manrique, mis compañeros y amigos. Gracias por ser mi apoyo en los momentos más difíciles de este proyecto. Su conocimiento y experiencia han sido fundamentales para llevarlo a cabo con éxito. Son un gran ejemplo de equipo y colaboración.

Abstract

The use of traditional models such as the modified Debye-Hückel model, the Pitzer model, MSE (Mixed-Solvent Electrolyte), or e-NRTL (Non-Random Two Liquid - Electrolyte) for predicting colligative effects in the Water + NaCl system is challenging. While these models have shown good results in terms of predictions, their statistical and computational implementation has required significant effort. On the other hand, certain Machine Learning algorithms have been studied for phase equilibrium prediction in systems with dissolved electrolytes. In this study, the implementation of three Machine Learning algorithms (Neural Networks, Least Squares Support Vector Machines, and Regression Decision Trees) was evaluated for predicting the decrease in melting temperature and saturation pressure of the Water + NaCl system. The results were compared with the prediction provided by an empirical variant of the Debye-Hückel model. Zero mean, normality, and residual independence tests were conducted for all models to statistically evaluate the regression results. It was found that machine learning models have the potential to predict colligative effects in electrolyte solutions, particularly the Regression Decision Tree model, which met all the assumptions studied for both effects and proved to be a reliable prediction tool. Finally, it was demonstrated that computationally, the implementation of machine learning models was straightforward, and their implementation for new studies in property prediction is a promising research area.

Keywords: Prediction, Colligative effects, Cryoscopic effect, Boiling point elevation, Water + NaCl, Empirical models, Debye-Hückel model, Machine Learning, Neural Networks, Least Squares Support Vector Machines, Regression Decision Trees, Electrolyte solution, Melting temperature, Saturation pressure.

Resumen

El uso de los modelos tradicionales como el modelo modificado de Debye-Hückel, el modelo de Pitzer, MSE (Mixed-Solvent Electrolyte) o e-NRTL (Non-Random Two Liquid - Electrolyte) para la predicción de los efectos coligativos del sistema Agua + NaC es díficil porque aunque han tenido buenos resultados en términos predicciones, su implementación de forma estadística y computacional ha requerido diferentes esfuerzos. Por otro lado, se ha estudiado la aplicación de algoritmos de Machine Learning para la predicción de equilibrios de fase en sistemas con electrolitos disueltos. En este trabajo se evaluó la implementación de 3 algoritmos de Machine Learning (Redes Neuronales, Máquinas de Soporte de Vectores de Mínimos Cuadrados y Árboles de Decisión de Regresión) para la predicción de la disminución en la temperatura de fusión y la presión de saturación del sistema Agua + NaCl. Los resultados se compararon con la predicción dada por una variante empírica del modelo de Debye-Hückel. Para todos los modelos se realizaron pruebas de media cero, normalidad e independencia de residuales con el objetivo de evaluar estadísticamente los resultados de regresión. Se comprobó que los modelos de aprendizaje de máquina tienen potencial para la predicción de los efectos coligativos de soluciones de electrolitos; especialmente se encontró que el modelo árbol de decisión de regresión cumplio con todos los supuestos estudiados para ambos efectos, y es una herramienta de precición fiable. Finalmente, se mostró que computacionalmente los modelos de aprendizaje automático fueron sencillos de implementar y que su implementación para nuevos estudios en la predicción de propiedades es un área de estudios prometedora.

Palabras clave: Predicción, Efectos coligativos, Efecto crioscópico, Efecto ebulloscópico, Agua + NaCl, Modelos empíricos, Modelo de Debye-Hückel, Machine Learning, Redes Neuronales, Máquinas de Soporte de Vectores de Mínimos Cuadrados, Árboles de Decisión de Regresión, Solución electrolítica, Temperatura de fusión, Presión de saturación.

Content

List of Figures

List of Tables

1 Introduction.

1.1 Justification.

The phase equilibrium models for electrolyte solutions are necessary for the design and thermodynamic prediction in industrial processes such as chemical fertilizer production, crystallization, wastewater treatment, reactive crystallization, azeotropic or extractive distillation, and liquid-liquid extraction [1]. There are many models of electrolytic solutions, but they are complicated, and it is difficult to choose which one to apply. On the other hand, it is necessary to critically compare the performance of the equations to predict the thermodynamic properties of electrolyte systems among themselves and against experimental data [2]. However, although these problems date back a few decades, the key points that currently hinder the application of these models are [3]:

- To choose the experimental databases according to the model application.
- Model comparison, since objective critical evaluations are rare and negative self-evaluations are even rarer.
- Implementation of published models, as they often hide details.
- Modifications or improvements lack proper justification and fail to distinguish between cause and effect.
- The need to consider multiple interactions between ions, which may require more complex models.

Examples of these problems have been found when attempting to apply models such as the Pitzer equation, MSE (Mixed-Solvent Electrolyte), or e-NRTL (Non-Random Two Liquid - Electrolyte). Furthermore, calculating properties without resorting to the optimization of specific parameters is particularly error-prone. In fact, *"all sorts of correlations have been described with non-trivial chemical systems, but the chances of these being universally applicable are therefore as bleak as they were a quarter of a century ago"* [3].

Artificial intelligence (AI), particularly machine learning (ML), can be as precise as current models using available computational resources such as processors, virtual machines, and others. Various review articles have been presented that include applications of AI/ML in chemical process engineering [4], fluid mechanics [5], energy systems [6], smart cities [7], and structural health monitoring [8]. In the field of petroleum engineering, some authors have demonstrated the validity of implementing these tools [9–13], while in the specific field of chemical engineering, interesting findings have been made on these applications [14–20]. However, to the best of our knowledge, the only ML application to the water-NaCl system has been made using support vector machines for the simulation of freezing point depression of the solution [21]. Therefore, it makes sense to pose the following research question: Is it possible to filter and reduce data related to colligative effects in the Water + NaCl system using a Machine Learning algorithm?

1.2 Hypothesis and objectives.

1.2.1 Hypothesis.

A Machine Learning model of the equilibrium data of the Water + NaCl system can predict its behavior, with statistical validity and with a prediction equivalent to that of the existing Debye-Hückel theoretical model.

1.2.2 General objective.

To develop a Machine Learning algorithm capable of filtering and reducing data related to colligative effects in the Water + NaCl system.

1.2.3 Specific objectives.

- Represent the experimental data of the NaCl-water system with an empirical variant of the Debye-Hückel (DH) model.
- To develop a computational reduction with ML capable of predicting the colligative effects of the water-NaCl system and its phase equilibrium.
- Statistically evaluate the performance of the ML model to reproduce the experimental data and compare it with the empirical model.

1.3 Methodology.

A compilation of published data on the cryoscopic and ebullioscopic effects in the Water + NaCl system was conducted to condense them into an empirical variant of the DH model. Data adjustment was achieved through a process of minimizing residuals.

Three different machine-learning models were evaluated: neural networks, support vector machines, and decision trees. The data used for training the ML models were exactly the same dataset used for the empirical variant. The concentration of NaCl (M_{MX}) was labeled as the input variable for the cryoscopic effect, and additionally, temperature (T) for the ebullioscopic effect; and as the output variables, melting temperature (T_{fus}) for the melting point prediction and saturation pressure (P_1^{sat}) for the boiling point elevation prediction.

Initially, a preliminary reduction of the complete dataset for each of the effects was performed using the four models under study. Pierce tests and Mahalanobis distance were used to filter outliers in the DH model, and the Minimum Covariance Determinant test was employed for each ML algorithm. Filtered data were used for the final model reduction and obtaining the presented results.

The validity of fitting processes for both the DH variant and ML models was assessed through residual diagnostics. Similarity, tests for normality like Shapiro-Wilk and Anderson-Darling, Durbin-Watson correlation, and the sign test for serial correlation were performed. Finally, graphs such as the QQ normal probability plot, parity diagrams, and the model-data comparison graph were presented.

2 Methods and models

2.1 Debye-Hückel model.

The first theoretical model of activity coefficients for electrolyte solutions was proposed by Debye and Hückel in 1923 [22]. It describes the behavior of the charged ions with the law of electrostatic charges, allowing the prediction of measurable effects such as freezing point depression [23]. In its original form, the Debye-Hückel model is limited to the study of solutions at infinite dilution, but a modified version including additional terms,

$$\ln(x_1\gamma_1) = -\upsilon(mw)_S M_{MX}\left[1 - \frac{\alpha}{3}\left|z^+z^-\right|\sqrt{I}\sigma(a\sqrt{I}) + \frac{\delta I}{2}\right], \tag{2-1}$$

can be applied to solutions with concentrations of practical interest [24]. Here the σ function is defined as

$$\sigma(x) = \frac{1}{x^3}\left[1 + x^2\ln(+x) - \frac{1}{1+x}\right] \tag{2-2}$$

and the ionic strength is

$$I = \frac{1}{2}\sum_{i=1}^{n} z_i^2 M_i = \frac{1}{2} M_{MX}\sum_{i=1}^{n}\nu_i z_i^2 \tag{2-3}$$

where n denotes the number of dissolved ionic species. For instance, for NaCl, $n = 2$, and thus, $I = M_{MX}$, while the remaining terms are shown in Table **2-1**.

In the particular case of the ebullioscopy effect the parameters a, α and δ of Eq. 2-1 were redefined as follows,

$$\alpha = a_1 + a_2 T \tag{2-4}$$

$$a = b_1 + b_2 T \tag{2-5}$$

Table 2-1: Aqueous NaCl parameters for the modified Debye-Hückel equation 2-1 [24].

Parameter	Symbol	Value	Units
Molecular weight	$(mw)_S$	0.01802	kg/mol
Ionic charges	$\left\|z^+z^-\right\|$	1	–
Stoichiometric factor	ν	2	–

$$\delta = c_1 + c_2 T \tag{2-6}$$

to improve the representation of the activity dependence on temperature (T).

2.2 Non-linear regression.

Most common regression models fit data to a straight line or a linear combination of independent variables. However, a regression can also describe the relationship between variables with non-linear functions, where the objective is the minimization of the sum of squares of the residuals (RES)

$$RES = \sum_{i=1}^{n} e_i^2 \tag{2-7}$$

to find, in an iterative optimization process, the values of the model parameters that best fit the experimental values. Where the residuals are defined as the difference between the observed values and the values predicted by the model

$$e_i = y_{i_{Exp}} - y_{i_{Cal}}. \tag{2-8}$$

In fact, this technique has been used to determine the parameters of nonlinear models in the analysis of liquid-liquid equilibria or electrolyte systems, with good experimental predictions [25, 26]. Currently, different optimization algorithms are used for minimization processes, among those are Newton's method, BFGS method, or Levenber-Marquartd method, which has been implemented and validated for structured nonlinear regression problems such as the settings performed in this work [27].

2.3 Machine learning.

Artificial Intelligence (AI) can be defined as the ability of digital computers to perform tasks that only humans could do until now. In particular, Machine Learning (ML) is an area of AI that aims to give computers the ability to "learn" a task without being explicitly programmed for it. Essentially, ML is a way of applying statistics to estimate complex functions and also, albeit with less emphasis, to obtain confidence intervals around them [28]. Machine learning is increasingly being used to predict the behavior of complex nonlinear systems in fields such as finance, medicine, geology, and sensing, among others. Machine learning algorithms, such as neural networks, decision trees, and support vector machines can predict performance and discern patterns that characterize a system by learning from data. They can also be used to model complex systems and automate the creation of analytical models. Machine learning models can be computationally fast and easier to implement compared to conventional thermodynamic models, for example in the prediction of clathrate hydrate equilibrium in the presence of electrolytes [29].

The development of an ML model includes the four critical steps shown in Figure **2-1** with these ingredients [30]:

1. **Database:** a collection of experimental and computational results that can be used to train and validate machine learning models. The quality of the data is a determining factor for the proper functioning of the model, so sometimes it is necessary to perform a previous cleaning, eliminating data that may be duplicated, irrelevant, or incorrect.

2. **Descriptors:** important data attributes used as inputs to the ML model. It is essential that the chosen descriptors are relevant to the objective result and are not highly correlated with each other.

3. **Algorithms:** mathematical models that predict a variable of a system based on other variables. They connect descriptors and qualitative and quantitative results, making their development one of the most active areas in machine learning. They can be grouped into two main categories: supervised learning and unsupervised learning. Supervised learning refers to using labeled data (known inputs and outputs) to train a model capable of predicting future input values (e.g., neural networks, Gaussian process regression, or support vector machines). Unsupervised learning uses unlabeled data to train the model and classify input data with little or no human intervention (e.g., k-means clustering, hierarchical clustering, Gaussian mixture model).

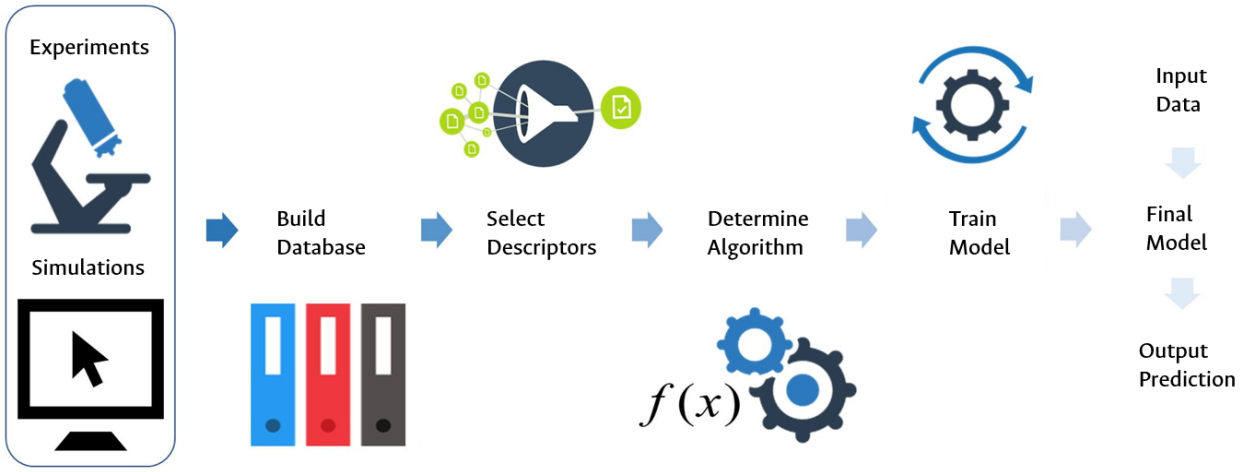

Figure 2-1: A general process for developing a machine learning model [30].

Neural network (NN).

It is a model, inspired by the connection of brain neurons, that is used to generalize the relationships that exist between inputs and outputs of a process. Classical neural networks are almost always presented as black-box models, fully data-driven, where the underlying principles of the model are generally absent. However, despite this nature, neural networks have been widely used in process optimization [31, 32], for the prediction of fluid properties of hydrocarbons, refrigerants, petroleum fluids, ionic liquids, and alcohols [33, 34], and the prediction of molecular properties [35]. In fact, neural networks have been shown to be universal approximates, due to their rich interpolation space, and therefore are a useful and promising tool for approximating physical laws [36].

An artificial neural network (NN) consists of a network of connected nodes called neurons organized into mutually exclusive layers, as shown in Fig **2-2**. The first is named the input layer, and it defines the independent variables of the network. The last one is named the output layer and returns the output values corresponding to the specified input variables, or the dependent or predicted variables. All intermediate layers are known as hidden layers [37].

In a Neural Network a neuron, shown in Fig **2-3**, is a processing element that receives a set of input signals $X = x_1, \ldots, x_n$, which are respectively modified by a series of synaptic weights $W = w_1, \ldots, w_n$. The values modified by the synaptic weights are summed up in what is called the *net input*. The output of the neuron depends on what we call the activation function, which acts on the net input [38]. Transfer functions are typically linear

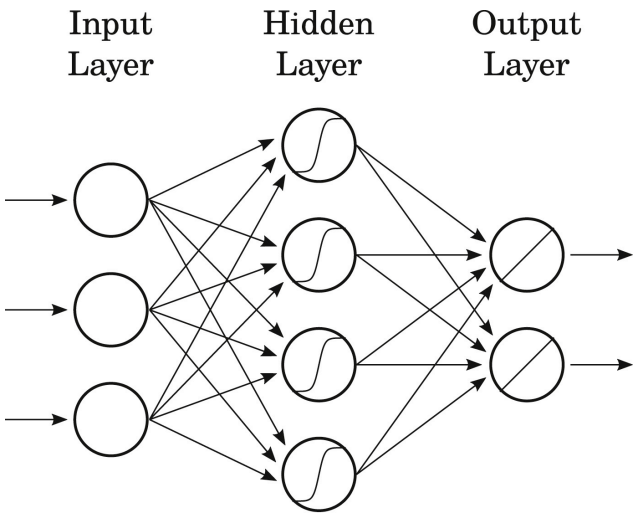

Figure 2-2: Schematic representation of a neural network with 3 layers, one input with 3 neurons, one output with 2 neurons, and one hidden with 4 neurons [33].

$$f(x) = x \tag{2-9}$$

in the output layer, and functions such as the sigmoidal hyperbolic tangent

$$f(x) = \frac{2}{1 + e^{-x}} - 1 \tag{2-10}$$

are used in hidden layers [39]. The setting of the number of hidden layers and neurons is commonly done through trial and error. However, the number of neurons in the input and output layers depends solely on the problem being investigated, and generally, those numbers are selected through optimization [39–42].

Once the architecture of a neural network has been finalized, the selection of an algorithm to train the model, i.e., to find the values of the weights that minimize the mean square error (MSE),

$$MSE = \frac{1}{n} \sum_{i=1}^{n} e_i^2, \tag{2-11}$$

is one significant stage of the optimization model. Although there are different algorithms to evaluate the model during the training process, the Levenberg-Marquardt algorithm has proven to be very competent, functional, with a high prediction capability, which makes it frequently used in the training of NNs [43–46].

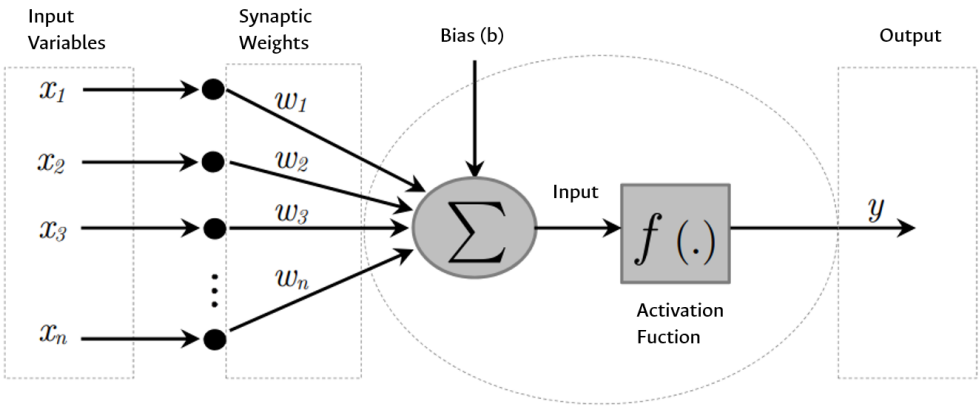

Figure 2-3: Neuron of an artificial Neural Network [38].

Least-squares support vector machine (LSSVM).

Support vector machines (SVM) are a well-accepted strategy for obtaining an accurate relationship between experimental data and the parameters of a particular mathematical problem [47, 48]. The Least-Squares Support Vector Machine (LSSVM) is a modified version of the support vector machine proposed by Suykens and Vandewalle [49]. This technique has been widely used for classification problems such as regression, reducing execution time, and increasing adaptability to different practical cases [50].

A typical SVM scheme is shown in Figure **2-4**, where input variables pass through functions called kernels. The results of these functions allow for determining the hyperplane that best fits the data. This procedure is carried out by minimizing the algorithm's cost function, defined as:

$$Q_{LSSVM} = \frac{1}{2}w^T w + Y\sum_{k=1}^{n} e_k^2 \tag{2-12}$$

for the LSSVM model, where w, Y, and e_k are the regression weight, the relative weight of the sum of regression errors compared to the regression weight, and the error for n training targets, respectively. The superscript T indicates the transpose matrix. Regression weight is written as

$$w = \sum_{k=1}^{n} \alpha_k x_k \tag{2-13}$$

and output as

$$y_k = w^T \phi(x_k) + b + e_k. \tag{2-14}$$

The feature map, the input vector of the model variables, and the outputs are connected by

$$\alpha_k = \frac{y_k - b}{x_k^T + (1/2\gamma)}. \tag{2-15}$$

The training process of the LSSVM technique involves adjusting the weights (w_i) and bias terms of the prediction function using an optimization algorithm that minimizes the objective function [21]. The kernel functions are essential for the performance of the technique and are used to project the data into a high-dimensional space. The most commonly used kernel functions in LSSVM are [50–52]

- Linear:

$$K(x_i, x_j) = x_i x_j. \tag{2-16}$$

- Polynomial:

$$K(x_i, x_j) = (x_i x_j + c)^d \tag{2-17}$$

  for $d \in \mathbb{N} \quad c \geq 0$.

- Gaussian (RBF):

$$K(x_i, x_j) = \exp\left(\frac{-\left\|x_i - x_j\right\|^2}{2\gamma^2}\right). \tag{2-18}$$

Currently, more emphasis has been placed on Gaussian radial basis kernels (RBF) due to satisfactory results with respect to the prediction of new values [9, 11, 12, 47, 49, 53–55].

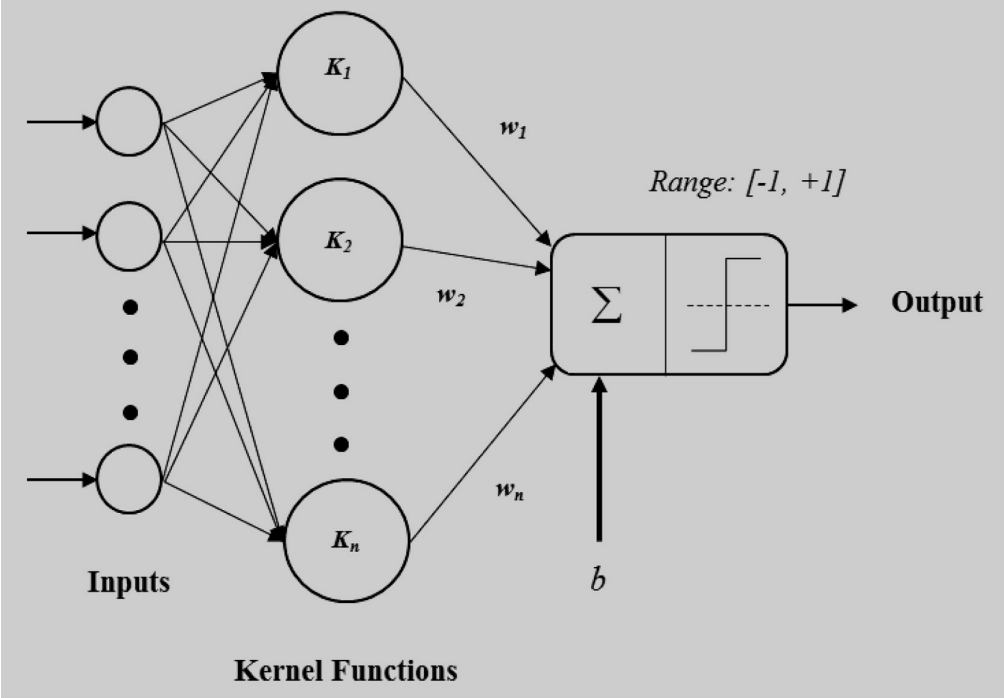

Figure 2-4: Typical schematic of a support vector machine (SVM) algorithm [21].

The parameters that affect the accuracy of the LSSVM models are [53, 56, 57]:

- The parameter C controls the margin violation penalty in the classification problem. The larger the value of C, the more strictly the margin violations are penalized and, therefore, the better the model fits the training data. This value is related to the first term of Eq 2-12 which corresponds to the penalty for the norm of the weight vector w.
- The γ parameter (Eq. 2-18) controls the shape of the Kernel function. A higher value of γ will result in a sharper Kernel function and thus a more complex model that is more prone to overfitting. A lower value of γ will result in a smoother Kernel function and a simpler model prone to underfitting.
- The parameter ϵ controls the tolerance of the loss function in the regression problem. If the absolute error between the model output and the desired output is less than ϵ, there is considered to be no error. A lower value of ϵ will result in a model that is a better fit for the training data.

The processes for the determination of the parameters in charge of the training of the model are governed by optimization decreasing the error of the adjustment [57].

Decision tree regression (DTR).

Decision trees (DT) are classification methods used in data mining and machine learning that predict the value of a dependent variable from given values of independent variables, as shown in Figure **2-5**. A decision tree classifies instances by descending down the tree from the root to some leaf node [58]. DT models are constructed using a hierarchical structure with multiple branches, the prediction task involves selecting a suitable branch based on the input conditions. Upon making a decision, the information passes onto the branches of the next level, and so on until the deepest level representing the final response to the problem is reached. The depth of the tree serves as a parameter for refining the model. Currently, several algorithms exist for training decision trees, which use the minimization of metrics like information entropy within various decision groups. In the case of regression trees (DTR), the metrics to be minimized correspond to the error between the experimental output and that obtained by the model [59–62].

All decision trees have the following elements [63]:

- **Node:** a position from which the tree will be split according to the independent variable and its value in the data set.
- **Edge:** is responsible for displaying the decision directly from one split to the next node.
- **Root:** the first node where the first division takes place.
- **Leaf node:** the final node that predicts the outcome of the model.

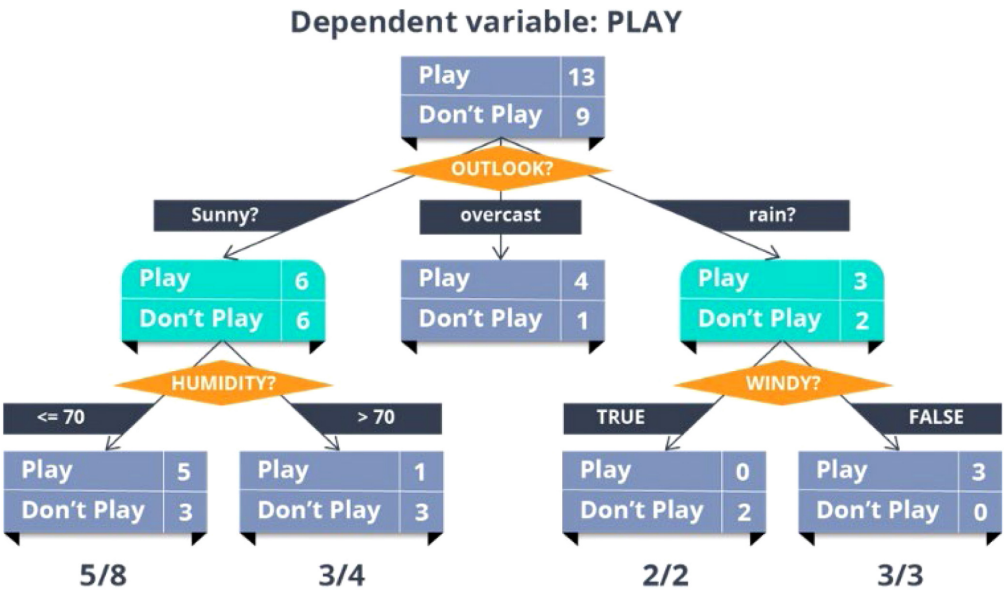

Figure 2-5: Illustrative diagram of a decision tree algorithm [64].

The two main aspects of algorithm attribute selection are:

- Gain of information: It is a method used to calculate the amount of information that a variable provides for the classification of a dataset. This method is used to split the tree into branches, based on the values of information gain, determined using a measure of impurity, which in the case of regression is the mean squared error (MSE). The idea is to select the variable that minimizes the MSE when dividing the dataset into two groups using that variable as a splitting node [63]. The procedure to determine this aspect is as follows:

  1. Calculate the MSE of the original data set.
  2. For each available variable, calculate the MSE of the subsets of data obtained by splitting the original set into two groups using the dependent variable as the splitting node.
  3. Calculate the information gain as the difference between the original MSE and the average MSE of the data subsets.
  4. Select the variable with the highest gain of information as the separating node.

  Once the splitting node has been selected, the process is repeated for the resulting subsets of data, until the stopping criterion is reached, which generally corresponds to the tree depth or minimum node size.

- Gini Index: Measures the impurity or purity defined as

$$G = 1 - \sum_i p_i^2 \tag{2-19}$$

where p_i corresponds to the ratio of the i class in the node [65]. This is used when creating the decision tree algorithm, where small Gini index attributes are preferred over attributes possessing a larger index. This parameter is calculated by the training algorithms and it is considered a classification or regression error [66].

2.4 Statistical consistency.

Outlier identification.

In the literature, outliers have been defined as «measurements that differ significantly from the normal pattern of the observer's data», although the reasons that can explain the occurrence of outliers are fundamental to data management, the selection and rejection of these values are extremely important for obtaining statistically valid predictive models [67, 68].

Various tests were conducted, to evaluate the quality of experimental residuals and to identify any possible outliers. Those tests have been used to improve models by measuring similarity in experimental samples and eliminating repeated values in cement content prediction processes. Additionally, it has also been used in biophysics studies to determine the importance of external features in statistical learning of human organ interactions [69, 70].

One of the tests used is the *Pierce* test, which aims to determine which outliers can be removed from the experimental data set. This test is based on the concept that if a value is an outlier, then its removal should significantly reduce the variance of the data set. The determination of outliers with this test is implemented in the following algorithm [71, 72]:

1. Calculate the mean ($\bar{x}$) and standard deviation (σ_x) of the dataset.

2. Determine the value of the statistic

$$R = \exp\left(\frac{1}{2}(x^2 - 1)\right)\psi\left(\frac{\sqrt{x^2}}{\sqrt{2}}\right) \tag{2-20}$$

where ψ correspond to the error function and x^2 to the squared maximun error deviation.

3. Evaluate
$$|x_i - \bar{x}|. \tag{2-21}$$
4. Remove those x_i such that
$$|x_i - \bar{x}| > R\sigma_x. \tag{2-22}$$
5. Repeat the procedure by incrementing the number of suspects by $+1$ and recalculating the value of R.
6. Recalculate the values of $\bar{x}$ and σ_x when no more suspects appear in the dataset.

Additionally, after determining the number of values to be rejected, the *Mahalanobis* distance measure can be used to find the values that are furthest from the covariance structure. This test is carried out through the following steps [73, 74]:

1. Calculate the mean ($\bar{\mathbf{x}}$) and covariance matrix (**COV**) of the dataset.
2. Calculate the Mahalanobis distance for each point in the dataset
$$D_i = (\mathbf{x_i} - \bar{\mathbf{x}})^\top \mathbf{COV}^{-1} (\mathbf{x_i} - \bar{\mathbf{x}})$$
where $\mathbf{x_i}$ is the feature vector of the i-th point.
3. Reject the number of outliers that have the largest distance from the determined structure.

Outliers can also be detected using the Minimum Covariance Determinant (MDC) estimator, calculated as the minimum determinant of the sample covariance matrix, and its Mahalanobis distance using the following algorithm [75, 76]:

1. Select a random subset of h observations from the dataset.
2. Calculate the covariance matrix from this subset.
3. Calculate the Mahalanobis distance (Eq 2) for each point in the dataset with respect to this covariance matrix.
4. Select the m observations with the smallest distances (where $m = \lceil n/2 \rceil$).
5. Calculate the covariance matrix from the selected m observations.
6. Calculate the determinant of this covariance matrix.

7. If the determinant is smaller than a value predefined by the confidence of the test, then the corresponding points are considered outliers.

The implementation of MDC has been compared with other tests for outlier detection in machine learning models, and its results have been satisfactory for its application in general cases [77]. Moreover, its ability to model data as a high-dimensional Gaussian distribution with potential covariances among input features allows this model to be used for anomaly detection and removal in machine learning applications. For instance, it has been employed for estimating performance and monitoring the status of thermal power plants, as well as detecting anomalies in the redshifts of SDSS galaxies [78, 79].

Statistical consistency.

In machine learning the data after outlier removal were used at the rate of 0.8 for training and 0.2 for testing [80]. The model loss and regression criterion were defined as the mean squared error (MSE Eq. 2-11).

The residuals' measures of central tendency were calculated as described in Table **2-2**, while the goodness of fit was tested using the residuals (Eq 3-12), to calculate the mean square error (Eq. 2-11), the coefficient of determination,

$$r^2 = 1 - \frac{\sum_{i=1}^{n} e_i^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2} \tag{2-23}$$

and the adjusted coefficient of determination

$$r^2_{adj} = r^2 - \frac{n-1}{n-(k+1)}(1-r^2) \tag{2-24}$$

where n and k are the numbers of experimental observations and model parameters, respectively, specifically for the case of ML models, the values of the adjusted coefficient were not determined due to the difficulty in obtaining the number of adjusted parameters.

The statistical consistency of the models is checked with tests for the following properties of the residuals [81]

- zero mean
- normality

Table 2-2: Definition of the statistics of the measures of central tendency for n residuals [82].

Statistic	Definition	Formula
AVG	Avarage of residuals	$\frac{1}{n}\sum_{i=1}^{n} e_i$
MAD	Mean average deviation	$\frac{1}{n}\sum_{i=1}^{n} \lvert e_i \rvert$
RMS	Root mean square	$\left[\frac{1}{n}\sum_{i=1}^{n} (e_i)^2\right]^{\frac{1}{2}}$

- independence (autocorrelation)

for instance, these assumptions validate the statistical consistency of liquid-vapor equilibrium models by reducing experimental data [82].

The Durbin-Watson test assessed their correlation [83, 84]. The statistic

$$d = \frac{\sum_{i=2}^{n} (e_i - e_{i-1})^2}{\sum_{i=1}^{n} (e_i)^2} \tag{2-25}$$

is compared with the critical values (d_L, d_U) as follows

- if $d < d_L$ there is evidence of **positive** correlation.
- if $d_L \leq d \leq d_U$ the test is inconclusive.
- if $d_U < d$ there is no evidence of **positive** correlation.

and the same is done, but using 4-d instead of d, for **negative** correlation [82]. However, the Durbin-Watson test has been modified for machine learning (ML) models, as it is difficult to predict the number of adjusted parameters. To do so, the test statistic has been analyzed based on the following criteria [85, 86]

- $d = 2$ zero autocorrelation.
- $d > 2$ **negative** autocorrelation.
- $d < 2$ **positive** autocorrelation.

Table 2-3: Normal distribution tests for residuals.

Test	Statistic	Formula
Shapiro-Wilk [82]	w	$\sum_{i=1}^{n}(a_i e_i)^2 / \sum_{i=1}^{n}(e_i - \bar{e}_i)^2$ $\bar{e}_i = \frac{1}{n}\sum_{i_1}^{n} e_i$
Anderson-Darling [91]	A^2	$-n - (1/n)\sum_i[(2i-1)\ln Z_i + (2n+1-2i)\ln(1-Z_i)]$

On the other hand, the randomness of sign grouping can be tested in terms of

$$p\pm = \frac{1}{C_m^{m+n}} \sum_{u=2}^{u'} f_u \tag{2-26}$$

being $p\pm$ the cumulative probability of observing at random u or less sequences of same-sign values given m positive and n negative values; high probabilities $p\pm$ suggest low autocorrelation, and vicev-ersa, and the values $p\pm\,(m,n,u)$ are tabulated in Ref [87]. The calculation of $p\pm$ in python has been automated with

$$f_u = 2C_{k-1}^{m-1}C_{k-1}^{n-1} \tag{2-27}$$

when $u = 2k$ or

$$f_u = C_{k-1}^{m-1}C_{k-2}^{n-1} + C_{k-2}^{m-1}C_{k-1}^{n-1} \tag{2-28}$$

when $u = 2k-1$. The use of $p\pm$ has been proposed before for the thermodynamic consistency eval-uation of liquid-vapor equilibria data [82], but anyway, it is limited to situations where the values follow a sequence, as in observations tied to a single independent variable. Since the prediction of the saturation pressure of the system depends on more than one predictor (molality and temperature), the order of the residuals was determined as a function of activity for both effects, a behavior that has been statistically validated previously [88].

In this work, the Shapiro-Wilk and Anderson-Darling tests were chosen to test the assumption of normality of the residuals. The effectiveness of these tests has been evaluated as a function of the number of experiments or data [89], and they have been used for the evaluation of models in vapor-liquid equilibrium [82], and even in data reduction for predictive models for COVID-19 cases [90]. Normality is assessed by the p-values of the tests at a specified significance level (%95).

3 Cryoscopic effect

3.1 Equilibrium model.

The cryoscopic effect occurs when a solute dissolves in a solvent, resulting in a decrease in the freezing temperature of the solvent. In other words, the cryoscopic effect is a type of solid-liquid equilibrium in which a solid is formed from the solvent due to the presence of the solute. Under isothermal conditions, for the solvent (1) the principle of equality of fugacities leads to [24]

$$f_1^S = \bar{f}_1^L \tag{3-1}$$

where the superscripts S and L correspond to the pure solid and fluid phases, respectively. Following the Lewis-Randall rule, the solubility of the solvent in the liquid phase is

$$\bar{f}_1^L = x_1 \gamma_1 f_1^L \tag{3-2}$$

where f_1^L corresponds to the fugacity of the pure liquid solvent. Substituting Eq. (3-1) into Eq. (3-2) the equality of fugacities in the system becomes

$$f_1^S = x_1 \gamma_1 f_1^L. \tag{3-3}$$

However, the solution temperature is far from the fusion temperature of pure solid, hence, the solid and liquid fugacities f_1^S, f_1^L do not cancel. Nevertheless, the ratio (f_1^L / f_1^S) is calculated as shown in Fig **3-1**, leading to [92].

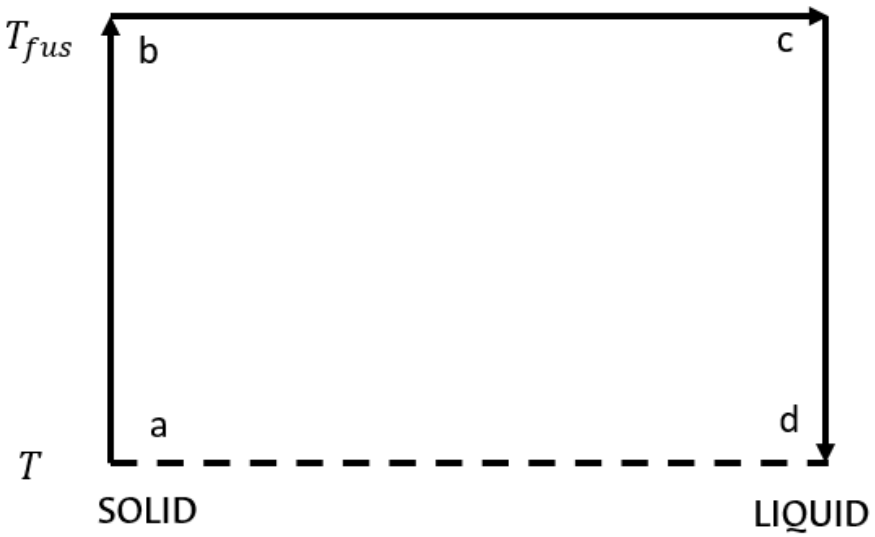

Figure 3-1: Thermodynamic cycle for the fugacity ratio of pure liquid [92]. The change between the points $a \rightarrow d$ is replaced with the sequence $a \rightarrow b \rightarrow c \rightarrow d$.

$$\Delta G_{a\rightarrow d} = \mathsf{RT}\ln\frac{f^L}{f^S} \tag{3-4}$$

where the molar Gibbs energy change, ΔG comes from the changes in enthalpy and entropy

$$\Delta G_{a\rightarrow d} = \Delta H_{a\rightarrow d} - \Delta S_{a\rightarrow d}\mathsf{T} \tag{3-5}$$

and following the cycle in Fig **3-1** the enthalpy change is written as

$$\Delta H_{a\rightarrow d} = \Delta H_{fus} + \int_{T_{fus}}^{T} \Delta \mathsf{C}_p d\mathsf{T}. \tag{3-6}$$

Similarly, the change in entropy over the thermodynamic cycle is written as follows

$$\Delta S_{a\rightarrow d} = \Delta S_{fus} + \int_{T_{fus}}^{T} \frac{\Delta \mathsf{C}_p}{\mathsf{T}} d\mathsf{T} \tag{3-7}$$

where the entropy of fusión is

$$\Delta S_{fus} = \Delta H_{fus}/T_{fus}. \tag{3-8}$$

Gathering Eqs. 3-5 to 3-8 and assuming that ΔC_p is constant over the range of temperatures in the cycle, it is obtained

$$\ln\frac{f^L}{f^S} = \frac{\Delta H_{fus}}{RT_{fus}}\left(\frac{T_{fus}}{T} - 1\right) - \frac{\Delta C_p}{R}\left(\frac{T_{fus}}{T} - 1\right) + \frac{\Delta C_p}{R}\ln\frac{T_{fus}}{T} \tag{3-9}$$

hence, from Eqs. 3-5 and 3-9

$$\ln(x_1\gamma_1) = -\left[\frac{\Delta H_{fus}}{RT}\left(1 - \frac{T}{T_{fus}}\right) + \frac{\Delta C_p}{R}\left(1 - \frac{T_{fus}}{T} + \ln\frac{T_{fus}}{T}\right)\right] \tag{3-10}$$

where

$$x_1\gamma_1 = \frac{f^S}{f^L}, \tag{3-11}$$

being $x_1\gamma_1$ the activity of component 1. Finally, Eq 3-10 allows determining the experimental activity for the determination of the residuals for the data reduction process.

To apply the Debye-Hückel model, we collected experimental data on the melting point depression (Fig. **3-2**) water + NaCl system, and performed a least-squares minimization of the sum of residuals (e_i), which were defined as follows:

$$e_i = (x_1\gamma_1)_{Exp} - (x_1\gamma_1)_{Cal} \tag{3-12}$$

where

- $(x_1\gamma_1)_{Exp}$ corresponds to the value of the activity calculated from experimental data.
- $(x_1\gamma_1)_{Cal}$ corresponds to the value calculates with Eq. 2-1 using the fitted parameters.

in this case, it was helpful to define the activity as $(x_1\gamma_1)$ and not with the regular notation a to avoid confusion with the fit parameter of the Debye-Hückel model.

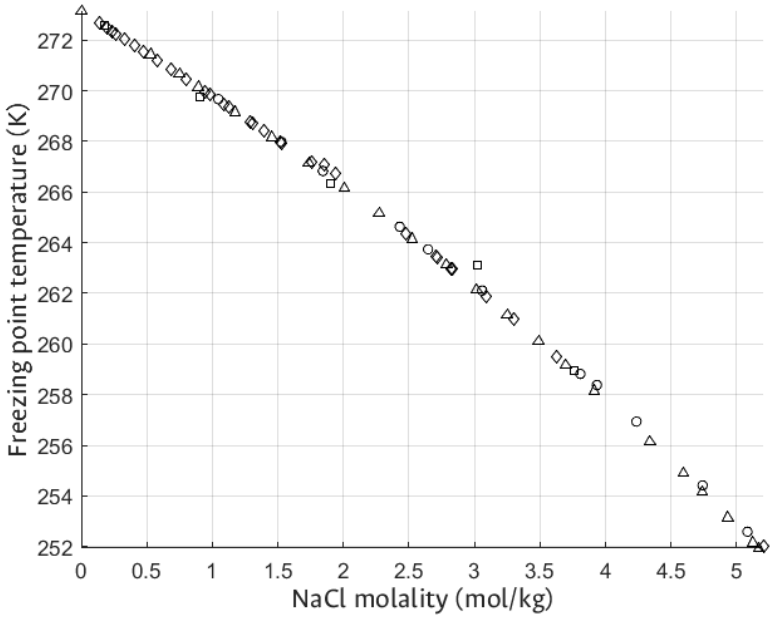

Figure 3-2: Freezing point experimental data for aqueous NaCl solutions. ○ [93], ♢ [94], △ [95], □ [96] .

3.2 Data filtering.

For this reduction, molality was the only independent variable. The Debye-Hückel parameter fit and Machine Learning algorithm training were first performed with the raw data set to determine the outliers from the residuals; that is, the differences between the experimental activities and their values calculated with the model, either DH or ML, as described in Sections 2.2 and 3.1. For the ML models, it means that the complete dataset was used to optimize the architectures or characteristic parameters.

The databases were defined as those used for DH, the algorithms used for effect prediction were the same as those presented in section 2.3 (Neural Networks, Leas Square Support Vector Machine, and Decision Tree Regression), and finally the predictor was defined as the only independent variable, the NaCl molality.

Debye-Hückel.

Debye-Hückel parameters for the Water + NaCl system (α, a, δ, Ec. 2-1) were fitted to raw data by residual minimization (Eq. 3-12), making use of the downhill simplex algorithm as implemented in the Python scipy.optimize.fmin function. Initial values were taken from Sandler's textbook [24], and the results are shown in Table **3-2**.

Neural network (NN).

The Keras libraries imported from Tensorflow in Python were used for training and construction of the algorithm architecture. Residual minimization was achieved by varying the neural network architecture. It was designed with one input variable, the molality of the NaCl (M), and one output, the fusion temperature of the system (T_{fus}). The optimal numbers of epochs (12) and neurons (1000) were found by a simultaneous cross-validation process, evaluating combinations of epochs (*100*, *500*, *1000*) and neurons (*4*, *8*, *12*). The model was trained with the *Levenberg-Marquardt* algorithm, which has shown promising results for modeling the vapor-liquid equilibrium of binary systems [46]. The number of hidden layers was increased one at a time until the mean squared error stopped decreasing (the MSE was used as the relationship between the experimental and calculated activities). Also, to avoid overfitting, the residuals' distribution in the QQ plots was checked, finding neither extrapolation problems nor a heavy-tailed distribution [97].

Least-squares support vector machine (LSSVM).

The Python-sklearn SVR library was used for the training and definition of the algorithm, which was applied by optimizing the margin penalty parameters (C), kernel function shape control (γ), and loss function tolerance used for regression (ϵ) through cross-validation. There were evaluated values of $C = [0.1, 1, 10, 100]$, γ =[0.001, 0.01, 0.1, 1, 10], and $\epsilon = [0.001, 0.01, 0.1, 1]$ [98–101], being choosen $C = 100$, $\gamma = 0.001$, and $\epsilon = 0.001$. The radial basis function, also known as the "RBF" or Gaussian function, was used as the kernel function, which has been previously used for predicting the freezing point depression in the NaCl + Water system [21].

Decision tree regression (DTR).

The Python DecisionTreeRegressor library from the Sklearn repository was used for this training. This model was generated by defining its depth through a cross-validation process, with depths ranging

Table 3-1: Outliers identified with each model.

Activity	M_{MX}	DH	NN	LSVM	DTR	Activity	M_{MX}	DH	NN	LSVM	DTR
0.9673	0.90			X		0.9053	3.02	X	X	X	X
0.9397	1.84				X	0.8855	3.30		X		
0.9420	1.86			X	X	0.8777	3.49		X		
0.9387	1.94			X		0.8062	5.20		X		
0.8963	3.01				X						

from 1 to 10. To avoid overfitting, the pruning parameter controlling the minimum reduction in cost-complexity was jointly optimized whit the depth of the tree. In this case, *ccpalpha* values ranging from 0 to 5 were evaluated [102]. The optimal parameters were 0 and 6 for the pruning control parameter and tree depth, respectively.

Outlier identification tests were applied to the raw data as described in section 2.4, Pierce and Mahalanobis tests for the DH residuals, and the MDC test for the ML models as explained in section 2.4. The results for both Debye-Hückel and Machine Learning, are shown in Fig. **3-3** and Table **3-1**. For example, only one outlier ($M_{NaCl} = 3.0195$ in Table **3-1**) was found in the DH residuals, both by Peirce and Mahalanobis tests. Also, it is worth mentioning that the QQ plots showed that indeed the selected values deviated from normality. On the other hand, the MDC test yields different outlier sets for each model, 4 outliers with NN, LSSVM, and DTR, a result that can be attributed to the various subspaces created in the test [103]. In the end, joining all the results, from DH and ML, 9 outliers were pruned from the 69 measurements of the raw data set, Table **3-1** shows a summary of the outliers rejected for each of the models.

3.3 Data reduction.

The model fitting and training processes were repeated after eliminating the outliers. The DH fitting was restarted using the parameters obtained from the raw data as new initial estimates, obtaining the parameters is shown in Table **3-2**. For neural networks, it was found that architecture with 2 hidden layers presented the best performance (see Table **3-3**). The goodness-of-fit statistics are summarized in Table **3-4**. Similarly, Figure **3-4** compares experimental data with model results (ML parity plots include only results of the test subset). For the machine learning, the data was partitioned at random, in training (80%) and test (20%, 12 measurements).

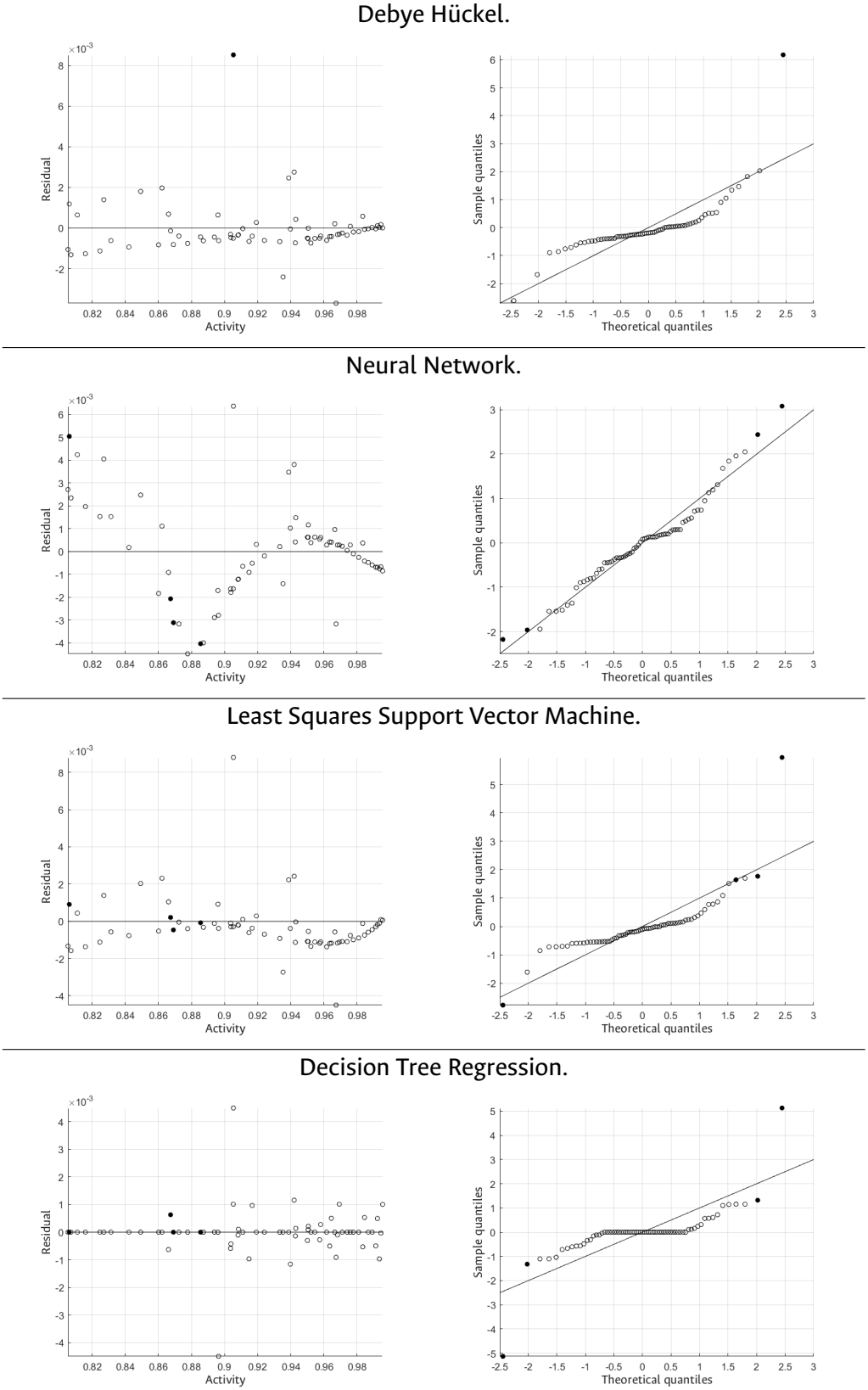

Figure 3-3: Residuals from the raw data. Bold symbols are outliers.

Table 3-2: Parameters of the Debye-Hückel model (Eq. 2-1). Filtered data excludes outliers (Fig **3-3**).

Parameter	Initial estimates	Raw data	Filtered data
α	1.178	1.9645	1.9576
a	1.4966	-4.2124×10^{-5}	-4.2938×10^{-5}
δ	0.137	2.5176×10^{-1}	2.5180×10^{-1}

Table 3-3: Neural network setting.

Hidden layers	**MSE**
1	1.2756×10^{-5}
2	9.0105×10^{-7}
3	1.3369×10^{-6}

Central tendency statistics show that the averages and mean absolute deviations (AVG, MAD, see Table **3-4**) of the residuals for all models tend to be zero. Moreover, compared to validated results of solubility equilibrium or molecular diffusivity, the values of the mean absolute deviation (MAD) are relatively small [84, 104, 105]. Values predicted with the DH model were in close agreement with the experimental results, as evidenced by the values of r^2 and r^2_{adj}. In the case of the ML models, the coefficient of determination shows a good correlation between the experimental data and the predictions for the NN and LSSVM models, but the DTR results show deviations with respect to some experimental data, an expected behavior due to the poor prediction capability of the model in regressions [106]. In sum, all these findings suggest that the models provide a good fit for the experimental data.

The distribution of the residuals and the correlation statistical tests are shown in Fig. **3-5** and Table **3-5**. The residuals are represented as a function of activity, this presentation with the objective of maintaining an order to analyze both colligative effects regardless of the number of independent variables as explained in the section 2.4. A negative correlation means that the probability of a residual of one sign being followed by another of the opposite sign is low. A positive correlation means a high likelihood that a residual will be followed by another of the same sign. The machine learning results come from the residuals of the test subset, and, as mentioned earlier (2.4) the criterion to analyze the

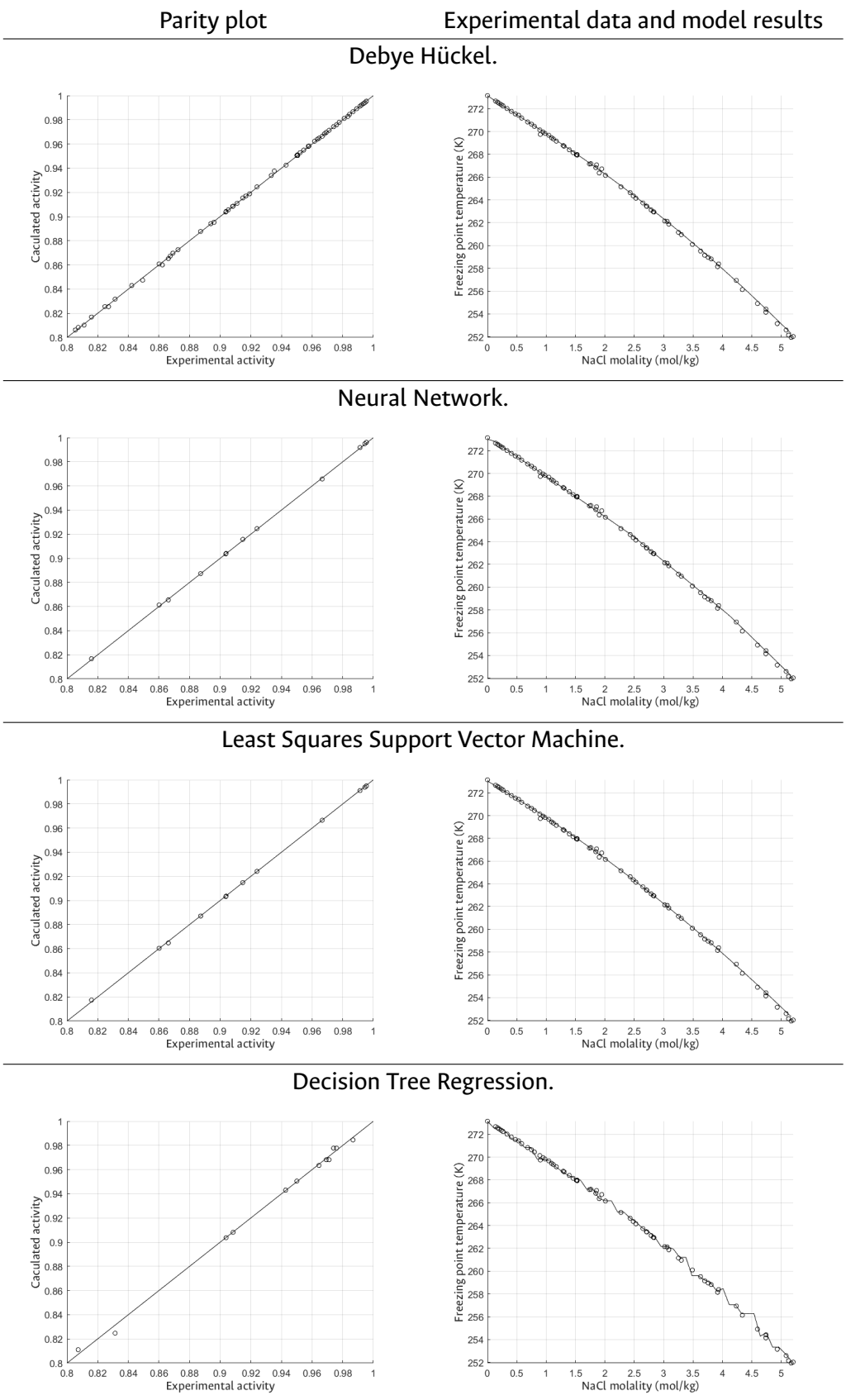

Figure 3-4: Model results with filtered data (ML parity plots include only results of the test subset).

Table 3-4: Residuals' central tendency and goodness of fit statistics by model.

	MSE	AVG	MAD	RMS	r^2	r^2_{adj}
DH	4.95×10^{-7}	-1.44×10^{-4}	5.05×10^{-4}	7.04×10^{-4}	0.9998	0.9997
NN	4.56×10^{-7}	-3.35×10^{-4}	5.93×10^{-4}	6.75×10^{-4}	> 0.9999	
LSSVM	3.95×10^{-7}	1.38×10^{-4}	4.36×10^{-4}	6.29×10^{-4}	> 0.9999	
DTR	7.72×10^{-6}	3.49×10^{-4}	2.00×10^{-3}	2.80×10^{-3}	> 0.9999	

d-value for ML models was different.

- For the Debye-Hückel residuals the plot suggests that both types of correlation appear, and in fact, the high $p\pm$ value indicates many sign sequences. However, the limit values of the Durbin-Watson test were $d_L = -4.73$ and $d_U = 1.86$, yielded evidence of negative correlation (YES for DW-), and no evidence of positive correlation (NO for DW+). The probability value calculated for the randomness of the distribution showed the same behavior so that the distribution of the residuals had no pattern.
- For the residuals of the NN and LSSVM models, a positive correlation was observed, with a non-random distribution, and few sign sequences (low $p\pm$).
- Contrary to the other two ML models, DTR generated residuals that are randomly distributed, with many sign sequences, and negative correlation.

To assess the normal distribution of the residuals, the *Shapiro-Wilk* and *Anderson-Darling* tests were applied to the filtered data as they serve equally well with small and large numbers of data [89]. The results of the two tests in Table **3-6** confirmed the normality of the residuals in Fig. **3-5** for all the models, that is, they follow the normal distribution according to the assumptions made. This behavior is visualized in the QQ plots in Figure **3-5**, The ML models present a distribution close to the 45-degree line, with no erratic behavior or deviation from the central data in the tails, and they don't show overfitting; which validates their use for further predictions. [97].

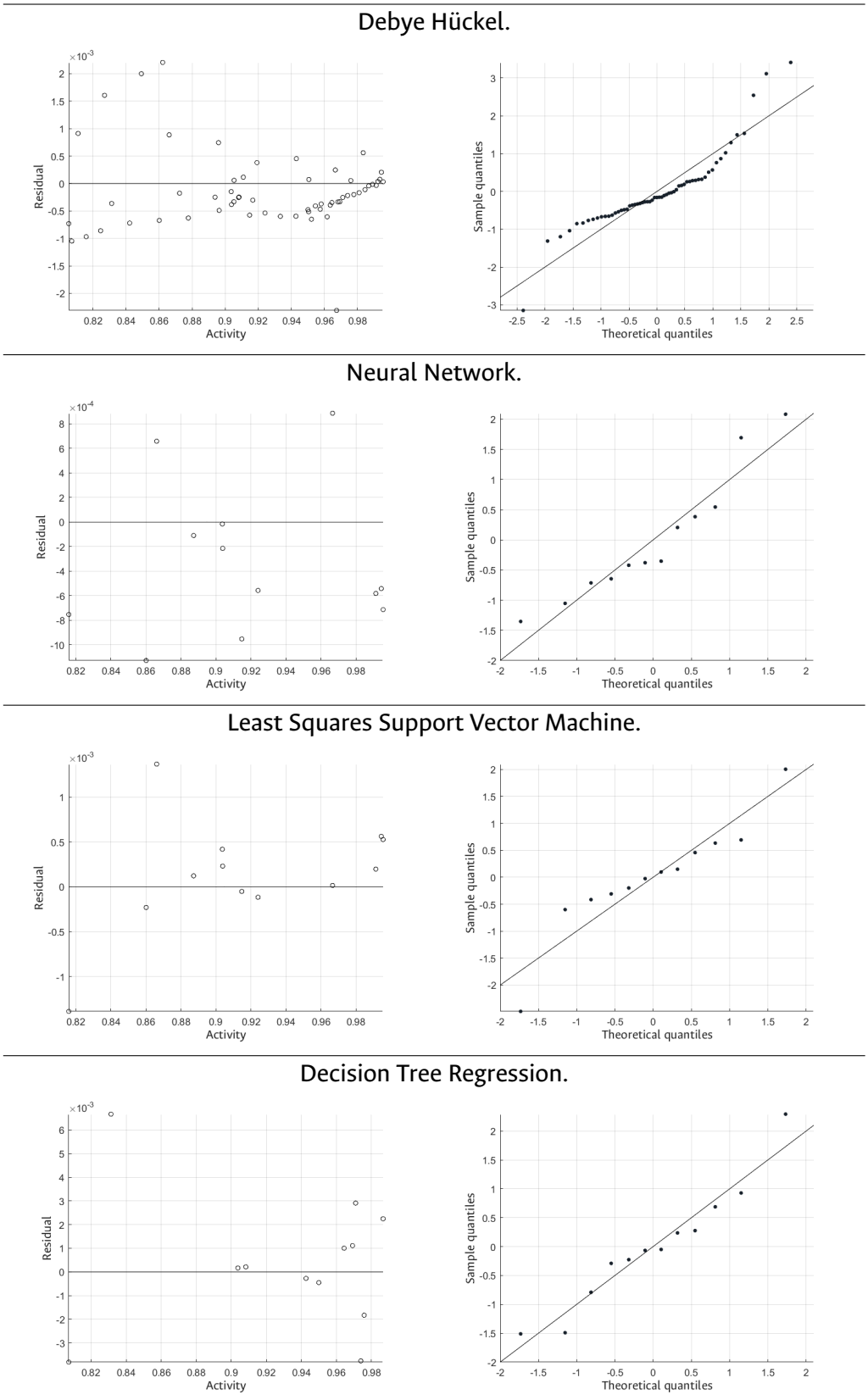

Figure 3-5: Distribution of residuals from the filtered data.

Table 3-5: Statistics and results of distribution tests for residuals.

Model	Durbin-Watson			Signs distribution				
	d-Value	DW-	DW+	$n+$	$n-$	Sequences	$p\pm$	Rand. Dist?
DH	*2.17*	YES	NO	*18*	*41*	*28*	*0.7688*	YES
NN	*1.63*	NO	YES	*2*	*10*	*3*	*0.1818*	NO
LSSVM	*1.23*	NO	YES	*8*	*4*	*4*	*0.1090*	NO
DTR	*2.40*	YES	NO	*7*	*5*	*5*	*0.4242*	YES

Table 3-6: Results of residuals' normality tests.

Model	Shapiro-Wilk		Anderson-Darling		Normal dist?
	Statistic	p-Value	Statistic	Critical value	
DH	*0.8684*	*1.2937*	-57.0280	*0.5430*	YES
NN	*0.9163*	*0.2273*	-10.0490	*0.4970*	YES
LSSVM	*0.9037*	*0.1768*	-10.0590	*0.4970*	YES
DTR	*0.9449*	*0.5636*	-10.0520	*0.4970*	YES

4 Ebullioscopic effect

4.1 Equilibrium model.

The ebullioscopic effect occurs when a non-volatile solute dissolves in a volatile solvent, which decreases the saturation pressure of the system. This effect can be considered as a type of liquid-vapor equilibrium between a gaseous phase composed of the solvent and a liquid phase formed by the solvent and the solute. In a system under isothermal conditions, the equal fugacity principle for the solvent (1) is expressed as [24]

$$\bar{f}_1^F = f_1^V, \tag{4-1}$$

and replacing the definition of the fluid fugacity (Eq 3-2) the system becomes

$$x_1\gamma_1 f_1^L = f_1^V, \tag{4-2}$$

where $x_1\gamma_1$ is activity of the solvent (1). For vapor-liquid systems, the vapor fugacity corresponds to

$$f_1^V = \phi_1 P, \tag{4-3}$$

where ϕ_1 is the fugacity coefficient of component 1, and the liquid fugacity, at low pressures, is expressed as [92].

$$f_1^L = P_1^{sat} \tag{4-4}$$

where P_1^{sat} refers to the saturation pressure of the pure component (1) at the system temperature. Finally, by replacing the fugacity definitions en Eq 4-2 one can express the activity of the system as,

$$x_1\gamma_1 = \frac{\phi_1 P}{P_1^{sat}} \tag{4-5}$$

but, at low pressures, the fugacity coefficient, ϕ, can be omitted, so Eq 4-5 reduces to [92]

$$x_1\gamma_1 = \frac{P}{P_1^{sat}}, \tag{4-6}$$

where the Antoine equation

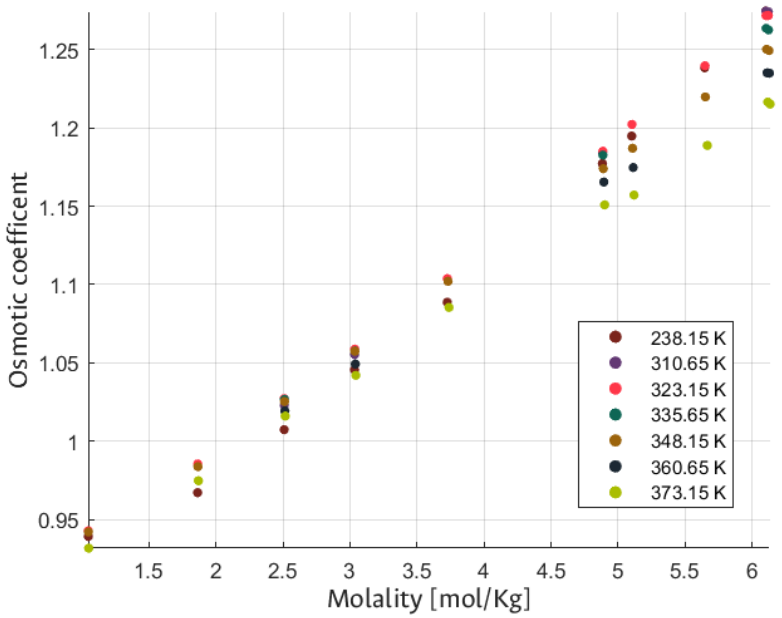

Figure 4-1: Experimental osmotic coefficients from ebulloscopic measurements in the NaCl + water system [109].

$$\log_{10}(P_1^{sat}[bar]) = 5.11564 - \frac{1687.537}{T[K] - 42.98} \tag{4-7}$$

was used to determine the saturation pressure of pure water as a function of temperature [107]. Many experimental activity measurements are reported in terms of the practical osmotic coefficient,

$$\Phi = -\frac{1000}{\upsilon M_{MX}(mw)_S}\ln(x_1\gamma_1), \tag{4-8}$$

introduced theoretically by Bjerrum [108]. For this work, all the experimental measurements found as osmotic coefficients or saturation pressures were converted into activities.

4.2 Data filtering.

We collected experimental data on the solvent saturation pressure decrease (Figs. 4-1 and 4-2) in the water + NaCl system, and performed a least-squares minimization of the sum of residuals (e_i Eq. 3-12) such as presented in the section 3.2

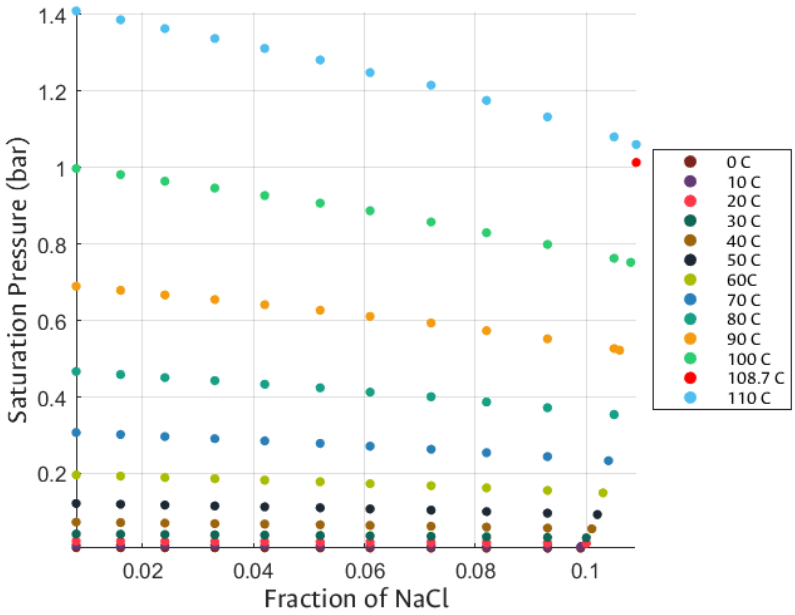

Figure 4-2: Experimental saturation pressures in the NaCl + water system [110].

The data filtering was performed similarly as shown in section 3.2, temperature was considered as an independent variable in addition to NaCl concentration.

Debye-Hückel.

Following Sandler's textbook a temperature dependence was included in the DH parameters [24]. In the data reduction to the equations [2-4 - 2-6] the initial estimates of $a1$, $b1$, $c1$ were the α, a, and δ values found in the previous chapter (3.2); while the temperature dependence coefficients ($a2$, $b2$, $c2$) were initialized as zero, the results are shown in Table **4-1**.

Table 4-1: Debye-Hückel parameters with T dependence (Eqs. 2-4, 2-5, 2-6) estimated from raw data.

Parameter	Value	Parameter	Value	Parameter	Value
a_1	-4.4838	b_1	-5.4753×10^{-1}	c_1	1.5466×10^{-1}
a_2	-1.8458×10^{-2}	b_2	7.6194×10^{-1}	c_2	-2.5196×10^{-4}

Neural network (NN).

The network was designed with two input variables corresponding to salt molality and temperature and one output variable corresponding to activity. The number of neurons (*7*, *13*, *20*) and epochs (*100*, *500*, *1000*) of adjustment were optimized using a cross-validation algorithm similar to that used in section 3.2 [111]. The reduction algorithm was specified as *Levenberg-Marquardt* [46]. As mentioned earlier, the number of hidden layers was increased one by one until the MSE stopped decreasing. The results showed that the best architecture for the neural network consisted of *13* neurons, and the optimal training was performed with *1000* epochs. Overfitting behavior was evaluated using QQ normal distribution plots [97].

Least-squares support vector machine (LSSVM).

The parameters corresponding to the margin penalty, the control of the kernel function shape, and the loss tolerance used in the regression were determined using a cross-validation algorithm, as described in section 3.2 [98–101]. The results after the optimization were $C = 10$, $\gamma = 0.001$, and $\epsilon = 0.001$. The radial basis function kernel was defined again as "RBF," which had already been used for this specific working system [21].

Decision tree regression (DTR).

This model received two input variables (molality and temperature), as proposed in the neural network model. The depth, which was the penalty parameter responsible for reducing the complexity cost, was determined through a cross-validation algorithm. Depth values between *1* and *15* and $ccp - alpha$ parameter values between *0* and *5* were tested [102]. The results obtained through cross-validation for the depth and the $ccp - alpha$ parameter were *9* and *0*, respectively.

For the DH model *Pierce* and *Mahalanobis* rejection tests resulted in no outliers or extremes in the raw data set. Figure **4-3** shows the machine learning outlier residuals determined by the MDC test for each model (see Table **4-2**). Once again, more outliers were identified for the ML models, due to the nature of the MDC test [103]. The QQ plots illustrate how they correspond to the most extreme values of the trend, so that after their removal the subsequent training will not be biased by these data. Gathering the results from the three models (NN, LSSVM, DTR) 26 outliers were removed from the 193 observations of the raw data set.

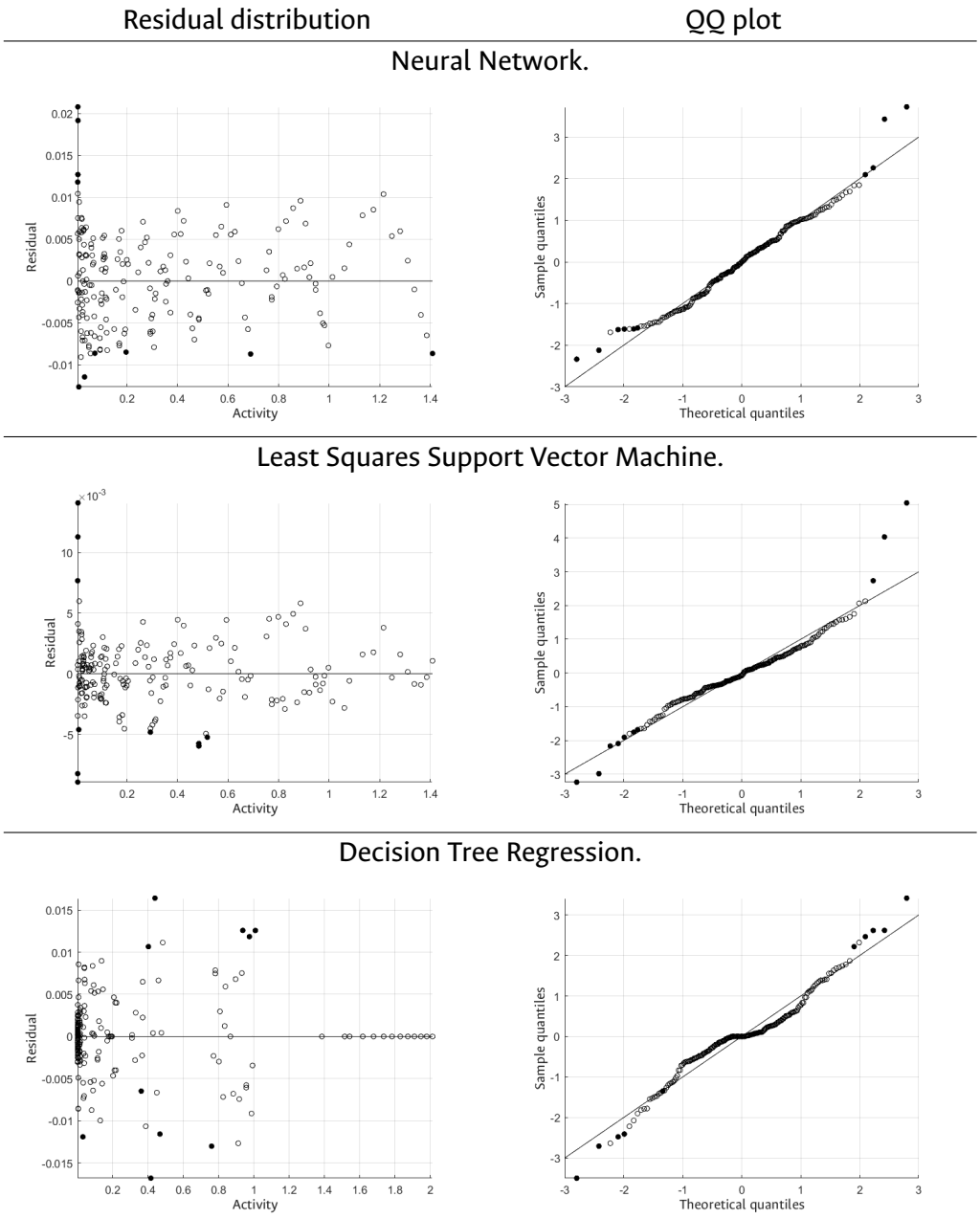

Figure 4-3: Residuals from the raw data. Bold symbols are outliers.

Table 4-2: Outliers identified with each model.

Activity	M_{MX}	T	NN	LSSVM	DTR	Activity	M_{MX}	T	NN	LSSVM	DTR
0.9703	1.36	273.15		X		0.9866	0.45	383.15	X		
0.9693	1.36	373.15			X	0.9869	0.45	373.15			X
0.9539	1.36	373.15			X	0.9852	0.45	363.15	X		
0.9527	1.36	363.15			X	0.9834	0.45	333.15	X		
0.9234	2.43	373.15			X	0.9820	0.45	313.15	X		
0.9210	2.43	273.15	X			0.8142	4.89	360.65		X	
0.8920	3.04	373.15			X	0.8059	4.96	273.15		X	
0.9045	3.04	273.15	X	X		0.7492	6.10	283.15	X	X	
0.8956	3.04	363.15			X	0.7618	6.11	360.65		X	
0.8767	3.41	363.15			X	0.7612	6.13	360.65		X	
0.8481	4.31	363.15			X	0.7441	6.72	373.15			X
0.7749	5.69	273.15		X		0.7483	6.37	333.15			X
0.7730	6.10	273.15	X	X		0.7489	6.17	303.15	X		

4.3 Data reduction.

After removing the outlier values the parameters of the Debye-Hückel model were set again by optimization (see Section 3.2) finding the results shown in Table **4-3**. After filtering the outliers, the ML models were also retrained. Different architectures of the neural network were evaluated with respect to the number of hidden layers with the MSE results shown in Table **4-4** that led to choosing 2 hidden layers. The results obtained for the goodness-of-fit statistics are shown in Table **4-5**.

Table 4-3: Debye-Hückel parameters with T dependence (Eqs. 2-4, 2-5, 2-6). Estimated from the filtered data set.

Parameter	Value	Parameter	Value	Parameter	Value
a_1	5.1244×10^{-2}	b_1	-9.8581×10^{-4}	c_1	-1.0583×10^{-2}
a_2	4.2250×10^{-4}	b_2	-6.4983×10^{-4}	c_2	3.9457×10^{-4}

The values of the AVG and MAD statistics are around zero, so it can be inferred that the goodness of fit

Table 4-4: Neural Network setting.

Hidden layers	MSE
1	2.6222×10^{-5}
2	6.9069×10^{-6}
3	2.7548×10^{-5}

Table 4-5: Residuals' central tendency and goodness of fit statistics by model.

	MSE	AVG	MAD	RMS	r^2	r^2_{adj}
DH	1.09×10^{-5}	1.00×10^{-3}	1.80×10^{-3}	3.30×10^{-3}	0.9980	0.9960
NN	6.91×10^{-6}	-1.30×10^{-3}	2.10×10^{-3}	2.60×10^{-3}	> 0.9999	
LSSVM	6.61×10^{-6}	3.92×10^{-4}	2.10×10^{-3}	2.60×10^{-3}	> 0.9999	
DTR	8.40×10^{-6}	1.20×10^{-3}	2.30×10^{-3}	2.90×10^{-3}	> 0.9999	

is adequate. The magnitude of the Mean Squared Error (MSE) showed that the models were trained adequately based on the error, i.e., all the models are consistent, and the assumption of mean deviation can be validated [112]. The results for the coefficient of determination r^2 for DH and ML models, and r^2_{adj} (only for DH), indicate a good prediction of the experimental values, also seen in Figure **4-4** for the vapor pressure of the mixture. On another hand, in the plots, the ML models did not show deviation from the experiment.

Fig. **4-5** shows the distribution of the residuals as a function of activity, with the objective of ordering the residuals as a function of a single variable instead of the two independent variables of the measurements (molality and temperature), Table **4-6** summarizes the statistics of the Durbin-Watson and sign distribution tests.

- For DH model comparison with the critical values of the Durbin-Watson test yielded an inconclusive result for positive correlation and no evidence of negative correlation. The result of the sign distribution test showed that the errors are not randomly distributed, due to the low

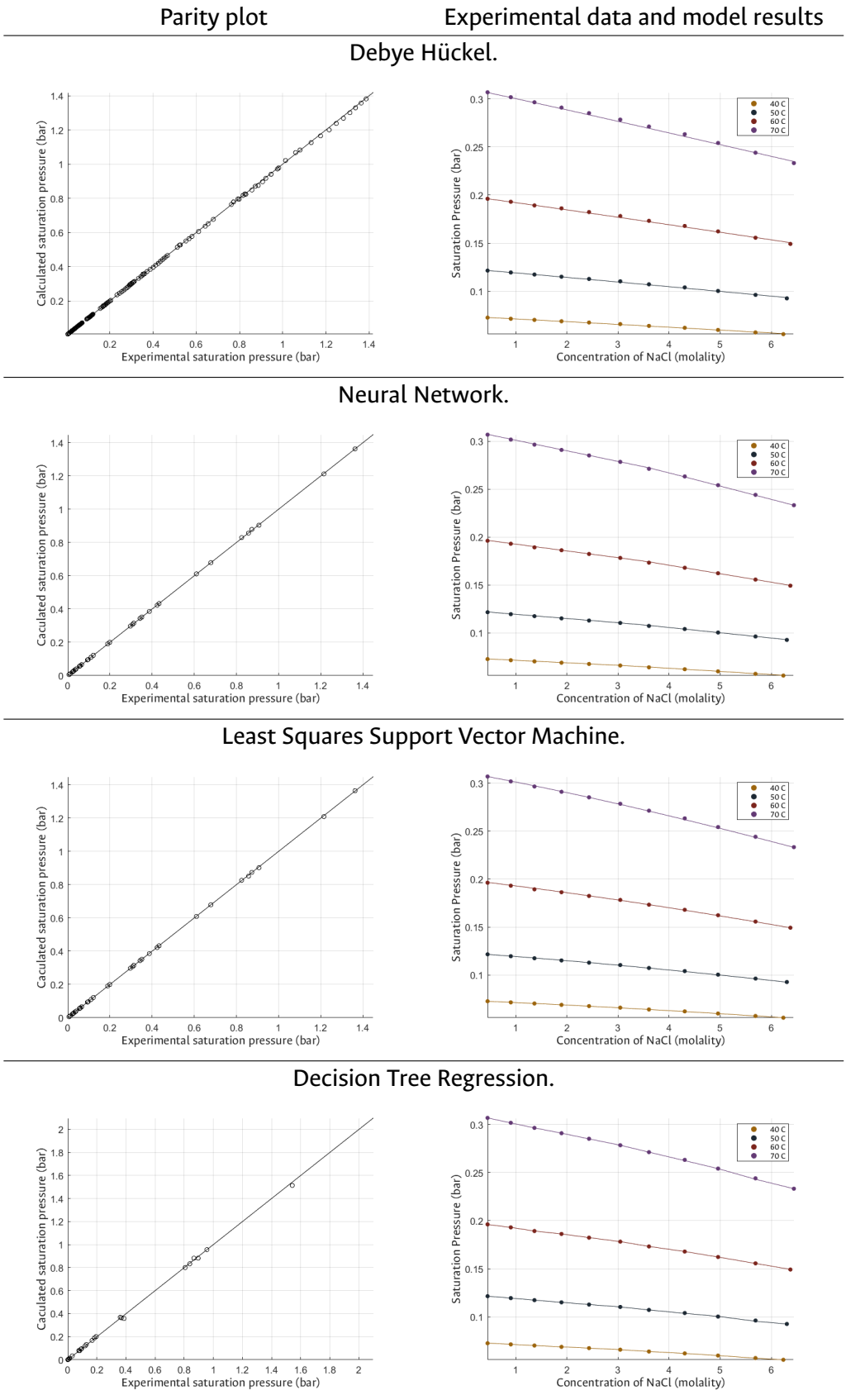

Figure 4-4: Model results with filtered data. The comparison of the model was done graphically at only a few temperatures to make the graphs clearer.

Table 4-6: Statistics and results of distribution tests for residuals. Where "?" is unconclusive.

Model	Durbin-Watson			Signs distribution				
	d-Value	DW-	DW+	$n+$	$n-$	Sequences	$p\pm$	Rand. Dist?
DH	0.83	NO	?	118	49	24	< 0.0001	NO
NN	1.42	NO	YES	9	25	9	0.0201	NO
LSSVM	1.68	NO	YES	17	17	16	0.3028	NO
DTR	1.72	NO	YES	24	10	17	0.8516	YES

probability determined.

- The results obtained for the Durbin-Watson test statistics showed that the residuals of the ML models were distributed following a positive correlation.
- The results of the randomness in the distribution of the residuals that were evaluated by the sign test showed that only the residuals of the DTR model were completely randomly distributed.
- That the Durbin-Watson results show a positive correlation may run counter to what was found for the randomness of the distribution of the residuals in the ML models. An explanation for this behavior with the results may be that the Durbin-Watson test is inaccurate in determining correlation in higher-order fitting models, so for the models worked, the result of the random test presents more solid results, even in large samples ($n > 100$), suggesting that the residuals of the evaluated models do not have any pattern in the way they are distributed and that this can be considered random [113, 114].

Similarly to Section 3.3 the results of the normality tests conducted for the models of ebullioscopic measurements are presented in Table **4-7** and Figure **4-5**, including the QQ normality plot. For the DH model, it is evident in the QQ plots that the residuals have a noticeably non-normal distribution. On the other hand, the other tests performed for the NN and LSSVM models allow us to confirm that the distributions of their residuals fit a normal distribution. For the DTR model, all tests and the QQ normality plot show significant deviations from the 45-degree line, indicating a significant deviation from the normal distribution.

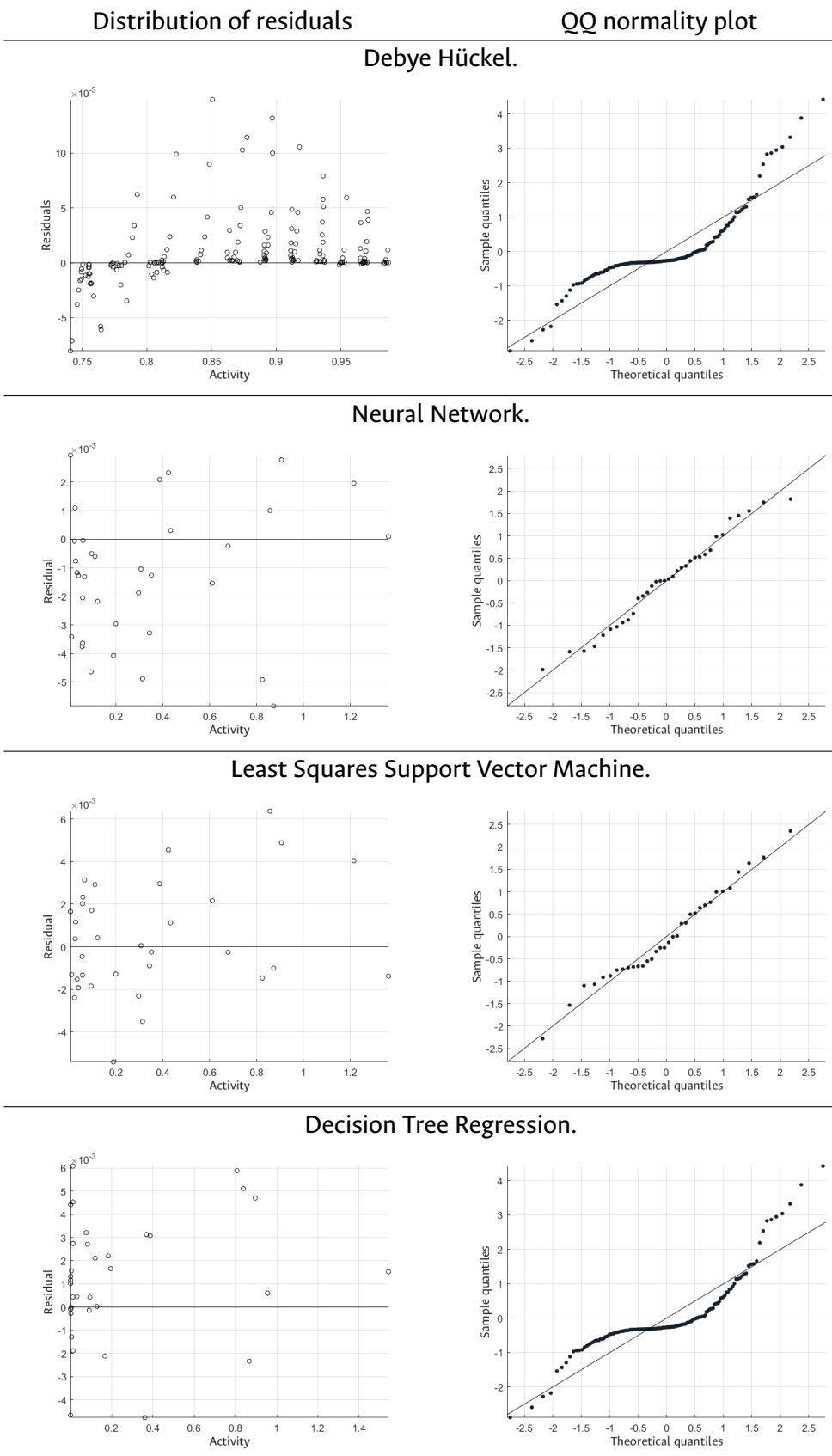

Figure 4-5: Distribution of residuals from the filtered data.

Table 4-7: Results of residuals' normality tests.

Model	Shapiro-Wilk		Anderson-Darling		Normal distribution
	Statistic	p-Value	Statistic	Critical value	
DH	0.7973	< 0.0001	13.3516	1.0670	NO
NN	0.9754	0.6247	-32.0140	0.5260	YES
LSSVM	0.9790	0.7397	-32.0170	0.5260	YES
DTR	0.9744	0.5942	-32.0180	0.5260	YES

5 Conclusions

The regression carried out on the experimental data yielded DH parameters different from those previously reported in the literature, however, these results were accepted, since they are consistent for both cryoscopic and ebullioscopic data. Nevertheless, the cryoscopic results presented better statistical consistency than the ebullioscopic ones, which that can be attributed to the dependence of saturation pressure on both molality and temperature.

It was found that the studied Machine Learning (ML) algorithms provide suitable predictions for the analyzed effects, although not all algorithms meet the valid statistical assumptions (zero mean, normality, and independence) for each effect. On the other hand, ML models are easier to create than the DH model. The comparison of the Debye-Hückel model and the algorithms (NN, LSSVM, and DTR) revealed that all of them have the potential to predict the two effects analyzed in this work.

All models adequately predict the freezing point reduction compared to the collected experiments (Fig. **3-4**). With respect to the sum of residuals (MSE), the DH model produced low values ($< 1 \times 10^{-}4$) for both cryoscopic and ebulloscopic effects. However, according to the measures of central tendency, the DTR model showed a better approximation to the prediction of the data, behavior similar to that shown by the NN and LSSVM models, showing that ML models have better predictive power for the applications studied here.

When comparing the regression models used, it was found that the DH model presented the highest probability of the residuals' sign distribution being random (no autocorrelation) for the cryoscopic effect, however, it did not have the same behavior when predicting the ebulloscopic effect where the residuals were notably dependent. Now, the lower values of the same probability for the NN and LSSVM models may suggest that their residuals do not follow the assumption of independence in both effects, cryoscopic and ebullioscopic. Although the probability of observing at random the residual sign distribution determined in the DTR model was not the highest, it shows an independent distribution of residuals for both the cryoscopic and ebulloscopic effects.

The evaluation of the normality showed that ML models yielded normally distributed residuals for both the cryoscopic and ebullioscopic data; while DH residuals followed the normal distribution for

cryoscopic results, but not for the ebullioscopic ones. Since the normality of the residuals is an important assumption for validating the predictive ability of the model, this suggests that the selection of an ML model over the theoretical DH model is statistically more accurate.

Finally, it has been statistically found that the DTR model performs better in predicting the colligative effects of the Water+NaCl system, as its central tendency measures, independence, and normality tests validated the zero mean, independence, and normality assumptions for the regressions and predictions.

Nomenclature table

Symbol

Cp	Heat capacity
G	Gibbs energy
H	Enthalpy
P	Pressure
S	Entropy
T	Temperature
$\bar{f}$	Partial molar fugacity
γ	Activity coefficient
e	Residual
f	Fugacity
γ	Fugacity coefficient
M	Molality (mol/kg)
R	Universal gas constant ($8.314 \quad J/mol \times K$)
x	Mole fraction
υ	Sum of the charges of the dissolved electrolyte
Φ	Osmotic coefficient

Subscript

1	Solvent
2	Solut
Exp	Experimental
Cal	Calculated
fus	Fusion
MX	Dissolved electrolyte

Superscript

S	Solid phase
F	Fluid phase
L	Liquid phase
V	Vapor phase
sat	Saturation

Acronyms

AI	Artificial Intelligence
ML	Machine Learning
DH	Debye - Hückel
NN	Neural Network
LSSVM	Least-squares Support Vector Machine
SVM	Support Vector Machine
DTR	Decision Tree Regression
DT	Decision Tree
MSE	Mean Square Error
MDC	Minimum Covariance Determinant
DW	Durbin - Watson

References

(1) Gómez G, M. Á.; Ibarra T, H. N., *Equilibrio de fases para sistemas electrolíticos.* Universidad Nacional de Colombia: Manizales, Col, 2015.

(2) Loehe, J. R.; Donohue, M. D. *AIChE Journal* **1997**, *43*, 180–195.

(3) May, P. M.; Rowland, D. *Journal of Chemical & Engineering Data* **2017**, *62*, 2481–2495.

(4) Lee, J. H.; Shin, J.; Realff, M. J. *Computers & Chemical Engineering* **2018**, *114*, 111–121.

(5) Brunton, S. L.; Noack, B. R.; Koumoutsakos, P. *Annual Review of Fluid Mechanics* **2020**, *52*, 477–508.

(6) Lund, H.; Østergaard, P. A.; Connolly, D.; Mathiesen, B. V. *Energy* **2017**, *137*, 556–565.

(7) O'Dwyer, E.; Pan, I.; Acha, S.; Shah, N. *Applied Energy* **2019**, *237*, 581–597.

(8) Flah, M.; Nunez, I.; Ben Chaabene, W.; Nehdi, M. L. *Archives of Computational Methods in Engineering* **2021**, *28*, 2621–2643.

(9) Eslamimanesh, A.; Gharagheizi, F.; Mohammadi, A. H.; Richon, D.; Illbeigi, M.; Fazlali, A.; Forghani, A. A.; Yazdizadeh, M. *Industrial & Engineering Chemistry Research* **2011**, *50*, 12807–12814.

(10) Gharagheizi, F.; Eslamimanesh, A.; Farjood, F.; Mohammadi, A. H.; Richon, D. *Industrial & Engineering Chemistry Research* **2011**, *50*, 11382–11395.

(11) Rafiee-Taghanaki, S.; Arabloo, M.; Chamkalani, A.; Amani, M.; Zargari, M. H.; Adelzadeh, M. R. *Fluid Phase Equilibria* **2013**, *346*, 25–32.

(12) Shokrollahi, A.; Arabloo, M.; Gharagheizi, F.; Mohammadi, A. H. *Fuel* **2013**, *112*, 375–384.

(13) Ghiasi, M. M.; Bahadori, A.; Zendehboudi, S. *Fuel* **2014**, *117*, 33–42.

(14) Kakkar, S.; Kwapinski, W.; Howard, C. A.; Kumar, K. V. *Education for Chemical Engineers* **2021**, *36*, 115–127.

(15) Ning, C.; You, F. *Computers & Chemical Engineering* **2019**, *125*, 434–448.

(16) Qin, S. J.; Chiang, L. H. *Computers & Chemical Engineering* **2019**, *126*, 465–473.

(17) Qin, S. J.; Dong, Y. *IFAC-PapersOnLine* **2020**, *53*, 11325–11331.

(18) Thon, C.; Finke, B.; Kwade, A.; Schilde, C. *Advanced Intelligent Systems* **2021**, *3*, 2000261.

(19) Udugama, I. A.; Gargalo, C. L.; Yamashita, Y.; Taube, M. A.; Palazoglu, A.; Young, B. R.; Gernaey, K. V.; Kulahci, M.; Bayer, C. *Industrial & Engineering Chemistry Research* **2020**, *59*, 15283–15297.

(20) Venkatasubramanian, V. *AIChE Journal* **2019**, *65*, 466–478.

(21) Yarveicy, H.; Moghaddam, A. K.; Ghiasi, M. M. *Journal of Natural Gas Science and Engineering* **2014**, *20*, 414–421.

(22) Debye V, P. *Phisikalische Zeeitschrift* **1923**, *24*, 185.

(23) Mourouga, G.; Chery, D.; Baudrin, E.; Randriamahazaka, H.; Schmidt, T. J.; Schumacher, J. O. *iScience* **2022**, *25*, 104901.

(24) Sandler, S. I., *Chemical, biochemical, and engineering thermodynamics*, Fifth edition; Wiley: Hoboken, NJ, 2017.

(25) Vasilyev, F.; Virolainen, S.; Sainio, T. *Chemical Engineering Science* **2018**, *175*, 267–277.

(26) Jaime-Leal, J.; Bonilla-Petriciolet, A.; Bhargava, V.; Fateen, S. *Chemical Engineering Research and Design* **2015**, *93*, 464–472.

(27) Johansson, R., *Numerical Python: Scientific Computing and Data Science Applications with Numpy, SciPy and Matplotlib*, 2nd ed; Apress: 2019.

(28) Machado Cavalcanti, F.; Emilia Kozonoe, C.; André Pacheco, K.; Maria de Brito Alves, R. In *Deep Learning Applications*, Luigi Mazzeo, P., Spagnolo, P., Eds.; IntechOpen: 2021.

(29) Acharya, P. V.; Bahadur, V. *Fluid Phase Equilibria* **2021**, *530*, 112894.

(30) Ding, J.; Xu, N.; Nguyen, M. T.; Qiao, Q.; Shi, Y.; He, Y.; Shao, Q. *Chinese Journal of Chemical Engineering* **2021**, *31*, 227–239.

(31) Koksal, E. S.; Aydin, E. *Computers & Chemical Engineering* **2023**, *174*, 108244.

(32) Manavi, S.; Becker, T.; Fattahi, E. *International Communications in Heat and Mass Transfer* **2023**, *142*, 106662.

(33) Poort, J. P.; Ramdin, M.; van Kranendonk, J.; Vlugt, T. J. *Fluid Phase Equilibria* **2019**, *490*, 39–47.

(34) Del-Mazo-Alvarado, O.; Bonilla-Petriciolet, A. *Fluid Phase Equilibria* **2022**, *561*, 113537.

(35) Rittig, J. G.; Ben Hicham, K.; Schweidtmann, A. M.; Dahmen, M.; Mitsos, A. *Computers & Chemical Engineering* **2023**, *171*, 108153.

(36) Masi, F.; Stefanou, I.; Vannucci, P.; Maffi-Berthier, V. *Journal of the Mechanics and Physics of Solids* **2021**, *147*, 104277.

(37) London, H. K. *Neural Networks, Springer-Verlag* **1996**.

(38) Cardona, A. C. *Inteligencia artificial* **2012**, 194.

(39) Khosravi, R.; Rabiei, S.; Bahiraei, M.; Teymourtash, A. *International Communications in Heat and Mass Transfer* **2019**, *109*, 104351.

(40) Tafarroj, M. M.; Mahian, O.; Kasaeian, A.; Sakamatapan, K.; Dalkilic, A. S.; Wongwises, S. *International Communications in Heat and Mass Transfer* **2017**, *86*, 25–31.

(41) Düğenci, M.; Aydemir, A.; Esen, İ.; Aydın, M. E. *Engineering Applications of Artificial Intelligence* **2015**, *45*, 71–79.

(42) Kurt, H.; Atik, K.; Özkaymak, M.; Recebli, Z. *International Journal of Thermal Sciences* **2008**, *47*, 192–200.

(43) Çolak, A. B.; Açıkgöz, Ö.; Mercan, H.; Dalkılıç, A. S.; Wongwises, S. *Case Studies in Thermal Engineering* **2022**, *39*, 102391.

(44) Ali, A.; Abdulrahman, A.; Garg, S.; Maqsood, K.; Murshid, G. *Greenhouse Gases: Science and Technology* **2019**, *9*, 67–78.

(45) Alrashed, A. A.; Karimipour, A.; Bagherzadeh, S. A.; Safaei, M. R.; Afrand, M. *International Journal of Heat and Mass Transfer* **2018**, *127*, 925–935.

(46) Carranza-Abaid, A.; Svendsen, H. F.; Jakobsen, J. P. *Fluid Phase Equilibria* **2023**, *564*, 113597.

(47) Chamkalani, A.; Mohammadi, A. H.; Eslamimanesh, A.; Gharagheizi, F.; Richon, D. *Chemical Engineering Science* **2012**, *81*, 202–208.

(48) Chamkalani, A. *Petroleum Science and Technology* **2015**, *33*, 31–38.

(49) Suykens, J. A. K.; Vandewalle, J. *Kluwer Academic Publishers* **1999**.

(50) Chamkalani, A.; Zendehboudi, S.; Chamkalani, R.; Lohi, A.; Elkamel, A.; Chatzis, I. *Fluid Phase Equilibria* **2013**, *358*, 189–202.

(51) Abe, S., *Support vector machines for pattern classification*; Springer: 2005; Vol. 2.

(52) Burges, C. J. C. *Data Mining and Knowledge Discovery* **1998**.

(53) Arabloo, M.; Shokrollahi, A.; Gharagheizi, F.; Mohammadi, A. H. *Fuel Processing Technology* **2013**, *116*, 317–324.

(54) Pelckmans, K.; Suykens, J. A. K.; Gestel, T. V.; Brabanter, J. D.; Lukas, L.; Hamers, B.; Moor, B. D.; Vandewalle, J. *Kasteelpark Arenberg 10* **2002**.

(55) Eslamimanesh, A.; Gharagheizi, F.; Illbeigi, M.; Mohammadi, A. H.; Fazlali, A.; Richon, D. *Fluid Phase Equilibria* **2012**, *316*, 34–45.

(56) Ford, D. M.; Dendukuri, A.; Kalyoncu, G.; Luu, K.; Patitz, M. J. *Computers & Chemical Engineering* **2020**, *141*, 106989.

(57) Serfidan, A. C.; Uzman, F.; Türkay, M. *Computers & Chemical Engineering* **2020**, *134*, 106711.

(58) Narayana Moorthy, N. S. H.; Martins, S. A.; Sousa, S. F.; Ramos, M. J.; Fernandes, P. A. *RSC Advances.* **2014**, *4*, 61624–61630.

(59) Di Caprio, U.; Wu, M.; Vermeire, F.; Van Gerven, T.; Hellinckx, P.; Waldherr, S.; Kayahan, E.; Leblebici, M. E. *Journal of CO2 Utilization* **2023**, *70*, 102452.

(60) Somvanshi, M.; Chavan, P.; Tambade, S.; Shinde, S. *International Conference on Computing Communication Control and automation (ICCUBEA)* **2016**, 1–7.

(61) Li, L.; Zhang, X. *International Conference On Computer Design and Applications* **2010**, 1, V1–155.

(62) Gupta, A.; Bansal, A.; Roy, K., et al. *5th International Conference on Intelligent Computing and Control Systems (ICICCS)* **2021**, 489–495.

(63) Singh Kushwah, J.; Kumar, A.; Patel, S.; Soni, R.; Gawande, A.; Gupta, S. *Materials Today: Proceedings* **2022**, *56*, First International Conference on Design and Materials, 3571–3576.

(64) Chauhan, V. K.; Shukla, S. K.; Tirkey, J. V.; Singh Rathore, P. K. *Journal of Cleaner Production* **2021**, *284*, 124719.

(65) Liu, X.; Zhang, J.; Pei, Z. *Progress in Materials Science* **2023**, *131*, 101018.

(66) Johnson, N.; Vulimiri, P.; To, A.; Zhang, X.; Brice, C.; Kappes, B.; Stebner, A. *Additive Manufacturing* **2020**, *36*, 101641.

(67) Karkouch, A.; Mousannif, H.; Al Moatassime, H.; Noel, T. *Journal of Network and Computer Applications* **2016**, *73*, 57–81.

(68) Al Samara, M.; Bennis, I.; Abouaissa, A.; Lorenz, P. *Journal of Network and Computer Applications* **2023**, *211*, 103563.

(69) Hao, X.; Zhang, Z.; Xu, Q.; Huang, G.; Wang, K. *Chemometrics and Intelligent Laboratory Systems* **2022**, *220*, 104461.

(70) Banus, J.; Lorenzi, M.; Camara, O.; Sermesant, M. *Medical Image Analysis* **2021**, *72*, 102089.

(71) Gould, B. A. *The Astronomical Journal* **1855**, *4*, 81.

(72) Ross, S. M. et al. *Journal of engineering technology* **2003**, *20*, 38–41.

(73) Mahalanobis, P. C. *Proceedings of the National Institute of Sciences of India* **1936**, *2*, 49–55.

(74) Beaulieu St-Laurent, P.; Gosselin, L.; Duchesne, C. *International Journal of Refrigeration* **2018**, *90*, 132–144.

(75) Rousseeuw, P. J.; Driessen, K. V. *Technometrics* **1999**, *41*, 212–223.

(76) Hardin, J.; Rocke, D. M. *Computational Statistics & Data Analysis* **2004**, *44*, 625–638.

(77) Fernández, Á.; Bella, J.; Dorronsoro, J. R. *Neurocomputing* **2022**, *486*, 77–92.

(78) Hundi, P.; Shahsavari, R. *Applied Energy* **2020**, *265*, 114775.

(79) Hoyle, B.; Rau, M. M.; Paech, K.; Bonnett, C.; Seitz, S.; Weller, J. *Monthly Notices of the Royal Astronomical Society* **2015**, *452*, 4183–4194.

(80) Tsutsui, K.; Moriguchi, K. *Calphad* **2021**, *74*, 102303.

(81) Drapper Norman R. Smith, H., *Applied Regression Analysis*, Third edition; Wiley: New York, NY, 1998.

(82) Carrero-Mantilla, J. I.; Ramírez-Ramírez, D. d. J.; Súarez-Cifuentes, J. F. *Fluid Phase Equilibria* **2016**, *412*, 158–167.

(83) Durbin, J.; Watson, G. S. In *Breakthroughs in Statistics: Methodology and Distribution*, Kotz, S., Johnson, N. L., Eds.; Springer New York: New York, NY, 1992, pp 260–266.

(84) Wisniak, J.; Polishuk, A. *Fluid Phase Equilibria* **1999**, *164*, 61–82.

(85) Kim, H. *Applied Economics* **2022**, *54*, 3197–3205.

(86) Singh, S.; Kulshreshtha, N. M.; Goyal, S.; Brighu, U.; Bezbaruah, A. N.; Gupta, A. B. *Journal of Water Process Engineering* **2022**, *50*, 103264.

(87) Swed, F. S.; Eisenhart, C. *The Annals of Mathematical Statistics* **1943**, *14*, 66–87.

(88) Draper Norman R.;Smith, H., *Applied Regression Analysis*, 3rd edition; Wiley: 2014.

(89) Mara, U. T. *Journal of Statistical Modeling and Analytics* **2011**.

(90) Uba, G.; Zandam, N. D.; Mansur, A.; Shukor, M. Y. *Bioremediation Science and Technology Research* **2021**, *9*, 13–19.

(91) D'Agostino Ralph B. Sthepens, M. A., *Goodness-of-fit techniques*; Marcel Dekker, INC: New York, NY, 1986.

(92) Prausnitz, J. M.; Lichtenthaler, R. N.; Azevedo, E. G. d., *Molecular thermodynamics of fluid-phase equilibria*, 2nd ed; Prentice-Hall: Englewood Cliffs, N.J, 1986.

(93) Rodebush, W. H. *Journal of the American Chemical Society* **1918**, *40*, 1204–1213.

(94) Gibbard, H. F.; Gossmann, A. F. *Journal of Solution Chemistry* **1974**, *3*, 385–393.

(95) Hall, D. L.; Sterner, S. M.; Bodnar, R. J. *Economic Geology* **1988**, *83*, 197–202.

(96) Haghighi, H.; Chapoy, A.; Tohidi, B. *Industrial & Engineering Chemistry Research* **2008**, *47*, 3983–3989.

(97) Gel, Y. R.; Miao, W.; Gastwirth, J. L. *Computational Statistics & Data Analysis* **2007**, *51*, 2734–2746.

(98) Li, G.-L.; Sato, O. *Acta Crystallographica Section E Crystallographic Communications* **2017**, *73*, 993–995.

(99) Zhang, N.; Yang, Z.; Chen, Z.; Li, Y.; Liao, Y.; Li, Y.; Gong, M.; Chen, Y. *Catalysts* **2018**, *8*, 246.

(100) Hsu, C.-W.; Chang, C.-C.; Lin, C.-J. *Department of Computer Science National Taiwan University*.

(101) Jabbar, M. A.; Deekshatulu, B.; Chandra, P. *Procedia Technology* **2013**, *10*, 85–94.

(102) Hastie, T.; Tibshirani, R.; Friedman, J., *The Elements of Statistical Learning*; Springer Series in Statistics; Springer New York: New York, 2009.

(103) AlMutawa, J. *Journal of Process Control* **2009**, *19*, 879–887.

(104) Descamps, C.; Coquelet, C.; Bouallou, C.; Richon, D. *Thermochimica Acta* **2005**, *430*, 1–7.

(105) Erdős, M.; Frangou, M.; Vlugt, T. J.; Moultos, O. A. *Fluid Phase Equilibria* **2021**, *528*, 112842.

(106) Kim, Y. S. *Expert Systems with Applications* **2008**, *34*, 1227–1234.

(107) Poling, B. E.; Prausnitz, J. M.; O'Connell, J. P., *The properties of gases and liquids*, 5th ed; McGraw-Hill: New York, 2001.

(108) Grundl, G.; Tsurko, E.; Neueder, R.; Kunz, W. *The Journal of Chemical Thermodynamics* **2019**, *139*, 105878.

(109) Gibbard, H. F.; Scatchard, G.; Rousseau, R. A.; Creek, J. L. *Journal of Chemical & Engineering Data* **1974**, *19*, 281–288.

(110) Engels, H., *Phase Equilibria and Phase Diagrams of Electrolytes*; Schön Wetzel GmbH: Frankfurt/-Main, FR, 1990.

(111) Hang, P.; Zhou, L.; Liu, G. *Journal of Cleaner Production* **2021**, *296*, 126615.

(112) Dohnal, V.; Fenclová, D. *Fluid Phase Equilibria* **1985**, *19*, 1–12.

(113) Dieudonné, N. T.; Armel, T. K. F.; Hermann, D. T.; Vidal, A. K. C.; René, T. *Technological Forecasting and Social Change* **2023**, *187*, 122212.

(114) Srianantha kumar, S. *Economic Modelling* **2013**, *33*, 126–136.